

Tietokonekulttuurin erikoislehti



Tee-se-itse  
raytracer

FLASH  
– kiitos ja  
näkemiin

Nettipokerin  
nousu ja tuho

ROBOLEMMIKIT  
ja virtuaalikumppanit

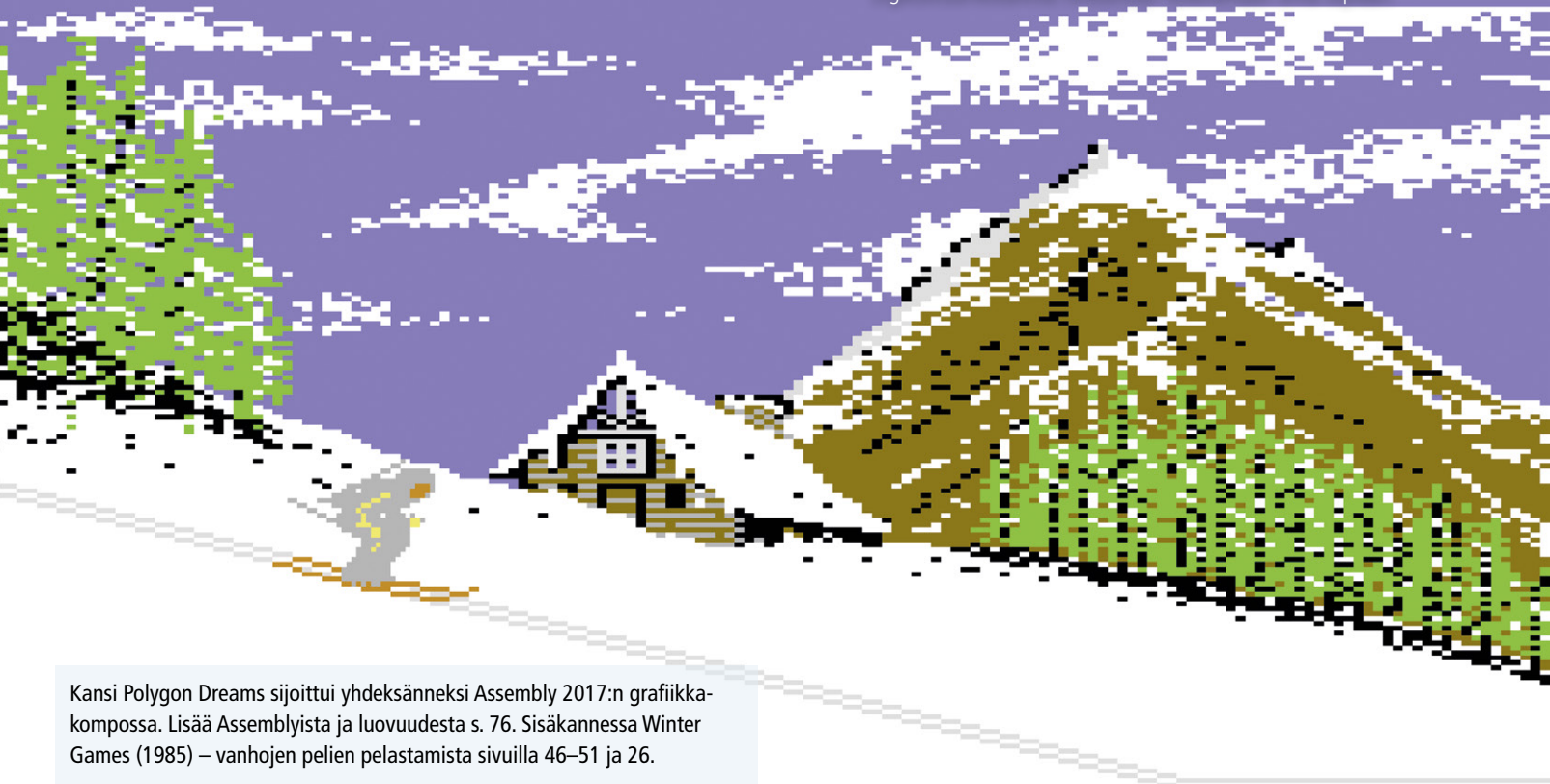
Langaton  
hiiri  
AMIGAAN

GAME MUSIC  
COLLECTIVE

KASSETTI-  
LAMERIT 2.0

FUNKTIONAALINEN  
OHJELMOINTI

- 3 Pääkirjoitus**
- 4 Flash**  
Muistokirjoitus rikkaalle, rakkaalle ja rikkinäiselle medialle.
- 9 Kolumni: Janne Sirén**  
Mikä on oikea ohjelmointikieli?
- 10 Tekoäly ei petä eikä jätä**  
Tietokoneohjelmasta elämänkumppani?
- 13 Robottilemmikit**  
OLED-silmät ovat sielun peili.
- 17 Vaarallisilla vesillä**  
Yksityisyydensuojan aakkoset.
- 21 Kolumni: Ronja Koistinen**  
Amatööri meni töihin.
- 22 Tee-se-itse raytracer**  
Renderöinnin opettelu voi aloittaa heijastavista palloista.
- 26 Lyhyet**  
Pikkujutuissa isoa asiaa – syväoppimista ja 8-bittisiä.
- 28 Nettipokerin lyhyt historia**  
Nettiuhkapelien villi länsi autioituu.
- 32 Koodaa oma pelitekoäly**  
Hyvä heuristiikka antaa pohjan vahvalle tekoälylle.
- 38 Funktionaalinen ohjelmointi**  
Koodia ilman sivuvaikutuksia.
- 44 Ei näin!**  
CD-levy on aika särkyvä. Siksi se kannattaa myös pakata huonosti.
- 46 Kasettilamereiden seurassa 2.0**  
Skrolli palaa digiarkeologian kaivauksille.
- 52 Curta**  
Mekaaninen taskulaskin syntyi keskitysleirillä.
- 55 Langaton hiiri Amigaan**  
Settlers sujuu parhaiten sohvalta.
- 60 Sarjakuva**  
Turrikaanien yö seikkailee siellä missä aurinko ei paista – Dungeon Masterissa.
- 62 Huuda kovempaa**  
Taipuuko kauhuelokuva tietokonepeliksi?
- 64 Se ainoa oikea peliohjain**  
Ammattilaistikon monet vaatteet.
- 66 Byterapers-graffiti**  
Demoryhmä iisalmelaisena kulttuuriperintönä.
- 69 Ihan pakko pelata**  
Täytyy pelata pelejä, jotta voi ostaa pelejä, jotta voisi pelata pelejä.
- 72 Game Music Collective**  
Pelimusiikki soi isossa konserttisalissa.
- 74 JOKstoriaa III**  
Kotimikroista kuultiin myös Kehä kolmosen ulkopuolella.
- 76 Assemblyt luovuuden alustana**  
Digikulttuurifestareilla rohkaistaan kokeilemaan omia siipiään.



Kansi Polygon Dreams sijoittui yhdeksänneksi Assembly 2017:n grafiikka-kompossa. Lisää Assemblyista ja luovuudesta s. 76. Sisäkannessa Winter Games (1985) – vanhojen pelien pelastamista sivuilla 46–51 ja 26.





Mikko Heinonen  
vastaava päätoimittaja

# Runsas jouluapaketti

Olemme Skrollissa pyrkineet siihen, että jokaisella numerolla on jonkinlainen löyhä teema. Tämä on numerosta riippuen näkynyt sisällössä enemmän tai vähemmän, mutta vuonna 2017 ehkä selkeimmin. Ensimmäisessä numerossa puhetta oli hakkeroinnista, toisessa keskiöön nousivat tietotekniikka-alan vaikuttajat ja kolmosessa taas lenneltiin hieman kevyemmissä merkeissä simulaatiotaivaalla.

Näin hieno ja johdonmukainen systeemi onkin aiheellista disruptoida välittömästi, joten joulunumeron teemaksi on taroituksella valittu ”vähän kaikkea”. Siihen johdattelee jo Antti Miettisen eli Dicen kansikuva, jossa näkyy monia skrollimaisia asioita.

Lehden johtojuttuna komeilee pitkään työstetty Flash-historiikki, jolla toivottavasti taputtelemme tämän verkkoa jo pitkään sulostuttaneen teknologian arvokkaasti autuaammille selaimenhidastusmaille. Lisäksi haluan nostaa esiin Miika Auvinen jutun virtuaalikumppaneista – tätäkin on pyriteltä toimituksessa pitkään, ja se käsittelee mielestäni harvinaisen tasapainoisesti aiheita, josta on vaikea kirjoittaa ylipäätään ja etenkin ilman moraalisaarvoja.

Iloinen olen myös siitä, että rautarakentelusta kirjoittajia alkaa löytyä miltei joka numeroon. Tällä kertaa tarjoilemme projektin Amigan hiiren muuttamisesta langattomaksi ja kutsumme myös lukijamme parantamaan prototyyppin toimivuutta. Vastaavia projektikuvauksia saa edelleen tarjota toimitukseen!

Jouluruljanssista selvittyään Skrollin toimitus ryhtyy valmistautumaan ensi vuoden lehtien tekoon. Sinä puolestasi voit halutessasi uusia tilauksen nyt vielä ”vanhalla” 35 euron vuosihinnalla osoitteessa [tilaaskrolli.fi](http://tilaaskrolli.fi). Tuttuun tapaan lähestymme teitä loppujakin alkuvuodesta sähköpostilla ja laskulla, mutta hinnassa on pientä korotuspainetta, joten nyt kannattaa napata ensi vuoden Skrollit plakkariin ajoissa. 🐿️

## Skrolli

Tietokonekulttuurin erikoislehti

**Yhteydenotot** toimitus@skrolli.fi  
Ircnet: #skrolli  
skrolli.fi

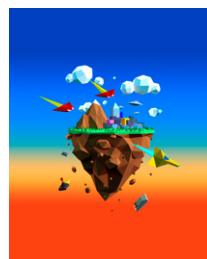
**Vastaava päätoimittaja** Mikko Heinonen  
**Päätoimittaja** Tapio Berschewsky  
**Toimituspäällikkö** Valhe Kouneli  
**Taiteellinen johtaja, taitto** Nasu Viljanmaa  
**Digipäälliköt** Toni Kuokkanen  
Janne Sirén  
**Taitto** Manu Pärssinen  
**Kuvatoimittaja** Laura Pesola  
**Mediamyynti** Jari Jaanto  
**Talous** Anssi Kolehmainen

**Muu toimitus** Jarno Niklas Alanko, Ville-Matias Heikkilä, Jukka O. Kauppinen, Ronja Koistinen, Sakari Lönn, Suví Sivulainen

**Tämän numeron avustajat** Heidi Akselin, Miika Auvinen, Robert Brotherus, Atte Haapalahti, Mikael Heikkanen, Ville Jouppi, Toni Kortelahti, Pinja Kuusela, Sakari Leppä, Teemu Leppänen, Tiina Malin, Antti Miettinen, Lasse Numminen, Jarkko Nääs, Lumi Pakkanen, Kimmo Rinta-Pollari, Aki Sivula, Benjamin Särkkä, Ismo Utriainen, Ilkka Wahala, Olli Wilkman

**Julkaisija** Skrolli ry

**Painopaikka** Hämeen Kirjapaino oy, Tampere,  
ISSN 2323-8992 (painettu)  
ISSN 2323-900X (verkkójulkaisu)



Etukannen kuva:  
Antti ”Dice” Miettinen



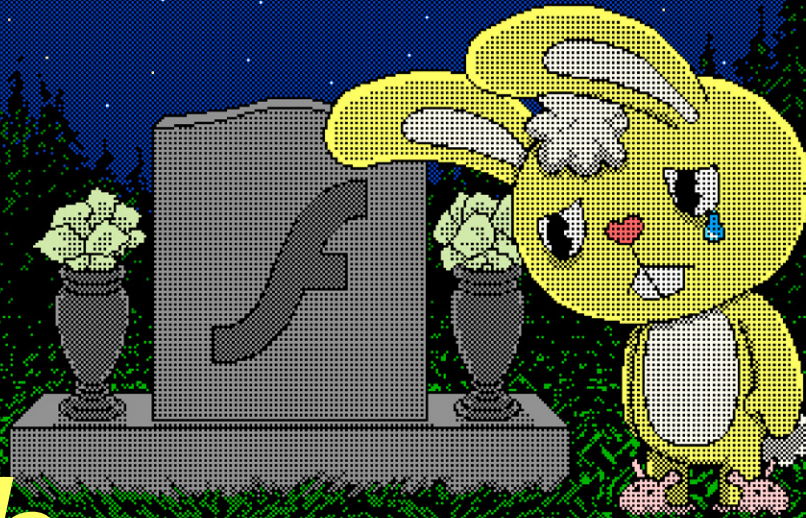
Sisäkannen kuva:  
Epyx, Janne Sirén



4041 0209  
Painotuote

HÄMEEN KIRJAPAINO OY





# Flash

## – erään aikakauden loppu

Teksti: Valhe Kouneli Kuvat: Ilkka Wahala, Valhe Kouneli

*Badger-badger, Sakarin villapaitapeli, The Ultimate Showdown (of Ultimate Destiny) ja miljoona erilaista selainpeliä, joita tuli klikseteltua yöt läpi vuosituhannen alussa. Ne kaikki tehtiin aikoinaan Flashilla, jonka kohdalla pitää paikkansa sanonta, että mitään ei voi todella kaivata, ennen kuin sen menettää. Flashin kulta-aika on ohi, ja vastaavaa aikaa tulee tuskin koskaan.*

**F**lash-teknologia mahdollisti helposti alustalta alustalle yhteensopivien pelien ja animaatioiden kehittämisen. Versiopäivityksestä toiseen ja käyttöjärjestelmältä toiselle pelit pelittivät ja kaikki oli paremmin kuin osasi edes arvostaa. Mutta aikansa kutakin, ja Adobe löi lukkoon tänä vuonna Flashin eläköitysmisvuodeksi vuoden 2020. Toisaalta Flash on tehnyt kuolemaa jo pitkään. Sen voi katsoa eläväksi kuolleeksi – niin jääräpäisesti se on pysytellyt mukana kuvioissa. Se jatkaa maallista matkaansa sitkeästi faniensa käsipuolella, ja ehkä sitä nähdään vuosikymmenienkin päästä demopartyjen Flash-kompoissa, jos se onkin jo saatu muualta kitkettyä.

Hyviksiä, pahiksia ja räjähdyskiä – vain yksi voi selvitä, kuka se on? Siitä on kyse The Ultimate Showdown of Ultimate Destinyssä.







We Drink Ritalin kuuluu animaation-tyyppisiin flash-videoihin. Niissä satunnaisesti tuntuvia kuvia näytetään usein sanoituksiltaan hämäräperäisen musiikin tahdissa.

Flash oli rakas ja luovuutta ruokkiva, mutta myös raskas ja tietoturvaongelmainen. **Steve Jobs** vetosi näihin syihin ilmoittaessaan, että Applen mobiililaitteet eivät tukisi Flashia. Se oli kuolettava isku. Voidaan spekuloida, oliko taustalla todellisuudessa henkilökohtainen kauna Adobea kohtaan tai kaupallinen intressi lisätä mobiilipelikehitystä ja -myyntiä, sillä vaikka Flashin ongelmat olivat todellisia, sen hyödyt olivat sivuuttamattomia. Mutta jätetään Steve-vainaja rauhaan ja laaditaan mieluummin kaunis nekrologi poistuvalla teknologialle.

### All your base, all your base...

Flash oli ja on matalan kynnyksen monityökalu. Sen elinkaari on ollut harvinaisen pitkä – yli kaksikymmentä vuotta – ja se on ollut koko ajan taaksepäin yhteensopiva aiempien versioidensa kanssa. Ei mikään huono saavutus internet-ajan teknologialta. Flashin alasarjasta huolimatta sitä käytetään yhä laajasti esimerkiksi animaatiotyökaluna. Televisiossa pyörivä lastensarja *Pipsa Possu* on tehty Flashilla, ja *My Little Pony – Friendship is Magic* hyödyntää laajasti Flashia tuotantoprosessissaan. Kilpailevia työkaluja on runsaasti, mutta Flashin yksinkertaisuus ja kätevyys pitää sitä elossa.

Useimmille Flash-animaatio tuo mieleen harrastelijoiden tekemät huumorivideot – Loituma-purjonpyörittäjä-animetyttö, *Hatten är din*, *The Llama Song*, Kikkoman-soijasupersan-

*Hatten är din*, hattu on sinun, lukee laulun sanoituksissa, mutta todellisuudessa laulun sanat ovat turkkia. Samasta väärin kuulemisen ideasta on revitty huumoria monessa muussakin Flash-videossa

kari, *We Drink Ritalin* ja *All your base are belong to us* -musiikkivideo...

Newgrounds, Albino Blacksheep ja Youtube olivat suosittuja, vapaita levityskanavia. Kun tekemisen kynnyks oli matala ja materiaalia julkaistiin paljon, suurin osa oli roskaa, mutta helmet roskan seassa sitäkin hienompia ja mieleenpainuvampia. Esimerkiksi ylitiösöpö mutta sitäkin verisempi *Happy Three Friends* aloitti nettijulkaisuna 1999, mutta sitä esitettiin lopulta televisiossakin.

### Mainokset ja mikro-ostot

Flash-pelien julkaisuun oli monia erilaisia väyliä, joista suosituin oli varmaan Newgrounds ja Suomessa ehkä Pelikone. Molemmat ovat yhä toiminnassa. Pelinsä tai muun flash-teoksensa pystyi lataamaan sivustolle ja jakamaan muille nopeasti. Käyttäjät kommentoivat sitä ja tekijä sai arvo-

kasta, välitöntä palautetta tuotoksestaan.

Aluksi pelejä tehtiin ja levitettiin hussin vuoksi, mutta pian joku näki markkinaraon ja pelien kylkeen ilmestyi mainoksia. Alan uranuurtaja oli Mochi Media, joka tarjosi pelinkehittäjälle työkalun, jolla peleihin saattoi lisätä mainoksia ja maksullisia lisäominaisuuksia. Alkoi mikromaksujen aikakausi. Mochi kokeili Flash-pelien yhteydessä mainostamista ja mikromaksuja ensimmäisenä isossa mittakaavassa ennen mobiilimarkkinoiden räjähdysmäistä kasvua. Mochia seurasi muun muassa Armor Games ja Kongregate. Niiden toimintamalli oli ostaa yksinoikeus pelin julkaisulle, tai että esimerkiksi jonkin aseensa sai vain, kun peliä pelasi yrityksen omalla sivulla.

Mainosten myyminen pelin oheen oli pelinkehittäjille jonkinlainen ansaintakeino. Jos peli menestyi isostiikin, saatettiin siitä kehittää kaupallinen versio jollekin toiselle alustalle ja laittaa Flash-versioon sen mainos. Parhaassa tapauksessa jokin iso teknologia- tai pelifirma palkkasi tekijät leipiinsä ja pelien parissa puuhastelusta tuli muutakin kuin harrastus tai sivuammatti.

Tavallaan Flash-pelien kehittäminen oli samanlaista kuin virtuaalitodellisuuspelien kehittäminen tänään. VR-markkinat ovat toistaiseksi niin pienet, ettei niille kehittämisestä ole suurta rahallista hyötyä, vaan monille se toimii lähinnä omien taitojen esittelyn alus-







Happy Tree Friendsissä on söpöjä otuksia ja pastellinen värimaailma mutta karut juonenkään-  
teet, joiden seurauksena söpöistä otuksista sinkoa verta ja suolenpätkiä.

tana ja mahdollisena panostuksena tulevaisuuteen. Joskus kehittäminen on vain kehittämistä omaksi iloksi tai itseilmaisun muoto.

### Yhteiskunnallinen kommentointi

Yksi uniikkeimmista Flashin mahdollistamista ilmiöistä oli nettikulttuuri, jossa pelillä saattoi kommentoida ajankohtaista tapahtuma tai ilmiötä. Kun uusin kohu-uutinen Britneystä tai Bushista tuli julki, aiheesta sikisi yksi jos toinenkin peli, joita oli hauska klik-  
sutella oman aikansa. Skriptaaminen

on nopeaa ja helppoa, joten tekemisen kynnyks oli matala. Yksinkertaista pelikonseptia pystyi kopioimaan yhtä helposti kuin nykypäivän kuvamakromeemejä, esimerkiksi kuuluisa Sakarin villapaitapeli poiki muun muassa kotimaisista uutisista inspiroituneet Vantaankosken kalapuikkopelin ja Annin pupupelin.

Toisaalta epäsuorasti Flashin ansiota on se, että Suomessa alettiin ensimmäisen kerran laajemmassa mittakaavassa keskustella tietokonepelien sisäisistä ostoista ja virtuaaliseen omaisuuteen liittyvistä lakiteknisistä kysymyksistä.

Kaikki aikaansa seuraavat nuoret koelivat nimittäin 2000-luvun alussa vähintään kerran Habbo Hotellissa chattailemista. Suomalaistiimin kehit-  
tämässä, alunperin Shockwavella tehdyssä mutta pian Flashille siirrettyssä palvelussa saattoi ostaa huonekaluja pelin sisäisellä valuutalla, jota puolestaan ostettiin oikealla rahalla joko netin kautta ja 2000-luvun alussa jopa R-kioskilta.

Ilmiöön liittyivät olennaisena osana kelmit, jotka kaappasivat toisten pelaajien hahmoja ja huonekaluja. Onko se varastamista, jos varastettua esinettä ei ole oikeasti olemassa? Kuinka järkevää on ylipääntään maksaa jostain eittodellisesta? Nykypäivänä nämä siihen aikaan ensimmäistä kertaa yleiseen keskusteluun nousseet kysymykset tuntuvat jo hieman kummallisilta.

### Kosketusnäytön kirous

Ennen älypuhelinien yleistymistä kaikki oli yksinkertaisempaa. Hiiri ja näppäimistö pysyivät samoina koneesta ja käyttöjärjestelmästä toiseen, mutta sitten kosketusnäyttö muutti kaiken. Netissä oleminen ei enää tarkoittanutkaan koneella istumista. Kevyt pelaaminen siirtyi mobiililaitteille, ja uusien pelien kehittäjillä oli harvoin motivaatiota yrittää ansaita elantoaan selainpeleihin kytkettävillä mainoksilla. Kolikon hintaiset mobiilipelit olivat joutopelien uusi normi.

Isoin muutos oli siinä, millaisia pelejä kehitettiin, koska kosketusnäyttö ra-



Alien Hominid oli ensimmäisiä Flash-pelejä, joka julkaistiin uudelleen konsolille isossa mittakaavassa. Pelin takana ovat Newgroundsin perustajat, ohjelmoija **Tom Fulp** ja animaattori **Dan Paladin**. Pelissä keltainen avaruusolio joutuu Maahan syöksytyään selviytymään korruptoituneiden FBI-agenttien hyökkäyksistä.



The Binding of Isaacista tehtiin myös kaupallinen, laajempi versio. Flash-versiossa kehoitetaan ostamaan peli Steamistä. Pelin juoni on inspiroitunut Raamatun tarinasta, jossa Jumala käskää Abrahamia uhraamaan poikansa merkinä uskollisuudestaan. The Binding of Isaacissa kaikki ei mene kuitenkaan niin hyvin kuin esikuvatarinassa, ja pelin kuva- ja äänimaailma tekevät siitä entistkin synkemmän.



Sakaran villapaitapeli.

joittaa pelissä käytettäviä kontrolleja. Sormen tarjoama tarkkuus on hiiren osoitinta rajallisempi, mutta toisaalta jotkin mekaniikat ovat luontevampia sormilla ohjattavaksi. Hiiren raahamista vaativat asiat, kuten ritsan venyttämisen ja laukaiseminen, ovat kuin tehtyjä kosketusnäytölle. Tarkkuutta vaativassa klikuttelussa taas hiiri voittaa nakkisormet ylivoimaisesti. Ei siis mikään ihme, että yksityiskohtaisia strategia- ja rakentelupelejä ei enää juuri tehdä kevytkoossa.

Kasuaalipelien julkaisukanavan vaihtuminen Flash-pelejä jakavasta nettikaupasta kännyköiden sovelluskauppoihin teki pelien julkaisemisesta myös hitaampaa. Julkaistu peli odottaa tarkastettavana nyt viikkoja, ennen kuin se pääsee Apple Storeen tai Google Playhin. Sovelluskaupasta laataminen on myös useamman askeleen homma kuin pikaviestimeen lähetetyn linkin avaaminen. Ajankohitaisiin asioihin reagoiminen hölmöllä pelillä ei ole enää nykypäivän ilmiö.

HTML5 toisaalta pyrkii Flashin korvaajaksi. Kun ensimmäiset uutiset Flashin kuolemasta saapuivat, HTML5 oli vielä lapsenkengissään, eikä siitä ollut vartenotettavaksi manttelinperi-jäksi. Se onnistui kuitenkin kehittymään vartenotettavaksi vaihtoehdoksi

parissa vuodessa, ehkä juuri kuolinuutisten kirittämänä. Näiden parin vuoden aikana ilmassa oli aistittavissa lievää paniikkia kehityspuolella. Monille pienille studioille HTML5-projektit ja käännökset johtivat tuotteen laadun heikkenemiseen ja moninkertaisiin kustannuksiin Flashiin verrattuna.

Toisaalta ajat ovat jo muuttuneet, ja vaikka HTML5 toteuttaisikin kaikki Flashin hyvät ominaisuudet, sille kehitettävät pelit suunnitellaan lähes aina myös kontrolleiltaan kännykälle sopiviksi. Älypuhelinien yleistymisen muutti pelintekokulttuurin pysyvästi, ja esimerkiksi hiiri/näppis-yhdistelmää käyttävät sekä millintarkkaa kur-



Crush the Castle tuntuu hieman vaikeamalta kuin sen innoittama Angry Birds, jossa ammuksen lentorata piirtyy näytölle ennen laukaisemista, eikä ajoitus ole niin tarkkaa.

sorilla tähtäämistä vaativat selainpelit ovat suurimmilta julkaisijoilta jääneet kokonaan pois.

### Parhaat pelit

Flashin kulta-aika oli kauneimmillaan ehkä 2010-luvun alussa, jolloin julkaistiin paljon huolella tehtyjä ja hitiksi nousseita pelejä. Flashilla oli helppo luoda yksinkertaisia pelejä, joten moni indie-pelistudio sai alkunsa Flash-peleistä. Esimerkiksi moderni tasehyppelyhitti *Super Meat Boy* oli alun perin Flashia, mutta siitä muovautui ajan mittaan täysimittainen peli, jonka voi ostaa nykyään niin Steamistä kuin konsolien latauskaupoista.

*Super Meat Boy*n lisäksi *vvvvvv*, *Super Motherload* ja koko tower defence- eli linnanpuolustusgenre olivat ensin Flashia. Itse *Angry Birds* sai inspiraationsa *Crush the Castle* -nimisestä Flash-pelistä, jossa tuhottiin katapultilla linnoja. Myös edelleen mobiilina klikuteltavien *Bejeweledin* ja *CandyCrushin* ensimmäiset versiot tehtiin selaimelle Flashilla. Moni nykypäivän pelinkehittäjä onkin aloittanut sillä uransa.

Ennen pelinkehittäjäksi ryhtyminen oli yksinkertaisempaa: aloita Flashilla. Nykyään pelialalle haluavalle on tarjol-

la monenlaista koulutusta, mutta työkalut ovat monimutkaistuneita ja erikoistuneita. Yhden kehittäjän projektit olivat vielä 2000-luvun alussa yleisiä ja kynnys peliprojektin aloittamiseen nykyistä matalampi.

## Mitä me emme jää kaipaamaan?

Kaikesta ihanuudesta huolimatta Flash oli ärsyttävä palikka, joka piti asentaa joka koneelle erikseen. Tai siis... Flash on edelleen ärsyttävä palikka, koska moni laajasti käytetty sivusto nojaa siihen toistaiseksi yhä. Sitä piti ja pitää päivittää säännöllisesti, mutta harva jaksoi – ja tässä piili iso ongelma. Peli- ja animaatiokäyttöön valjastettuna Flash pysyi vielä aisoissa, mutta selaimen kaatavat, Flashilla tapetoidut design-nettisivut eivät olleet täysin positiivinen ilmiö. Onneksi traumat eivät ole niin pahat, etteikö Zombo.com hymyilyttäisi yhä.

Emme kuitenkaan jää kaipaamaan surkeita, Flashilla tehtyjä sarjatuotantopelejä, joissa saman palapelipohjan päälle vaihdetaan aina uusi kuva, tai kuramerta, jonka seasta helmet piti kaivaa. Valitettavasti Flashin kuolema ei jätä tätä ilmiötä kokonaan taakseen, sillä kiinalaiset sarjatuotantopelit menevät varmasti sinne minne rahastamismahdollisuudetkin.

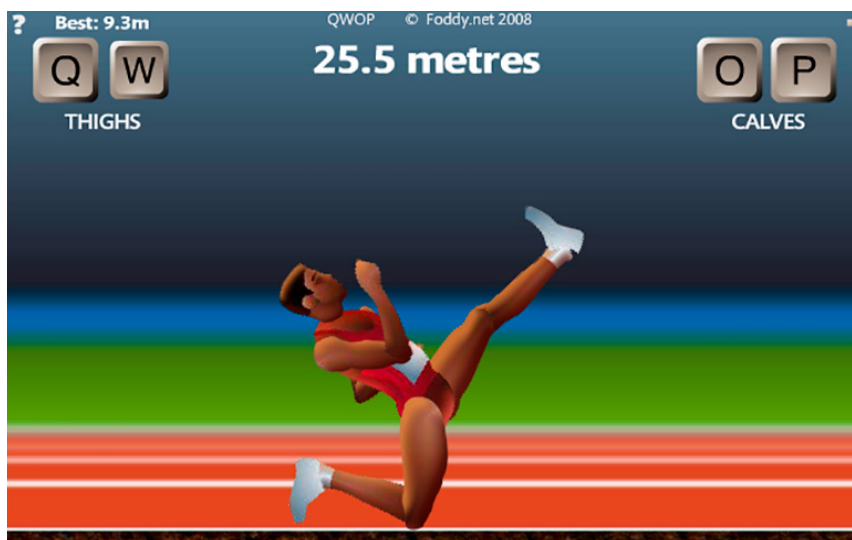
Kaikkialla tuntuu toteutuvan sama sykli. Uusi julkaisukanava tai -media syntyy, ja urheat pioneerit tarttuvat toimeen ja kehittävät mielenkiintoista sisältöä hyvillä tuotantoarvoilla ja luovuus kukoistaa. Hiljalleen kanava tai media yleistyy, lisää tekijöitä saapuu ja rahakin liikkuu. Tällöin haaskalinnut saapuvat paikalle.

Kunkin median kulta-aika on rajallinen, ja siitä kannattaa nauttia niin kauan kuin sitä kestää. Esimerkiksi virtuaalitodellisuuden yleistymisen on sellainen juttu, jonka pääsee kokemaan juuri nyt. 🐞

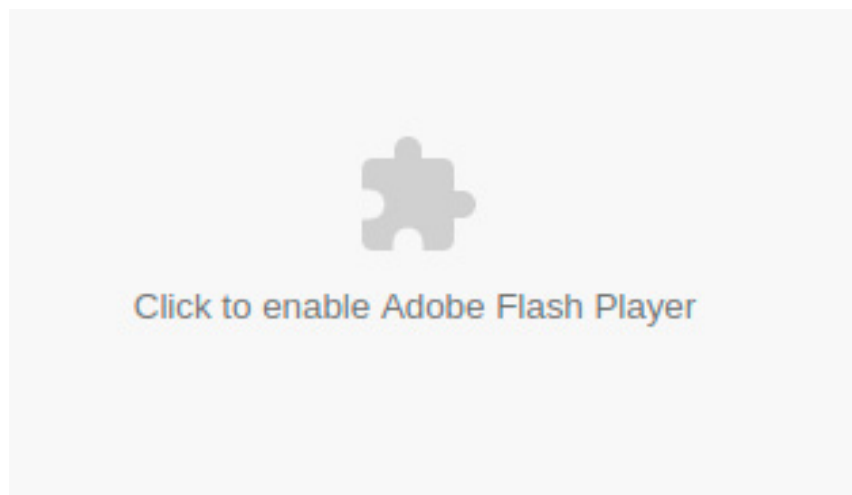
Juttua varten on haastateltu pitkän uran Flash-pelien parissa tehnyttä Pelikoneen **Thomas Arentoa**.



Meat Boy, Super Meat Boy'n Flash-edeltäjä, ei ollut ehkä ihan yhtä hienon näköinen, mutta kyllä siinäkin verinen liha lätsi seinä vasten loikkiessa.



Yritin ihan oikeasti, mutta sadan metrin juokseminen QWOPissa ihan vain tätä artikkelia varten oli liikaa vaadittu.





# Lelukieli

*Python ei ole oikea ohjelmointikieli.*

Janne Sirén

Julkaisimme viime numerossa artikkelin leluohjelmointikieli Scratchista (Skrolli 2017.3). Lelukieleksi kutsutaan yleisesti ohjelmointikieltä, jolla kielen erityisluonteen, esimerkiksi helppokäyttöisyyden takia, harjoitellaan tai kokeillaan ohjelmointia, mutta jolla ei sen rajoitusten tai puutteiden vuoksi yleensä toteuteta tuotantosovelluksia. Lelu, ei siis *oikea* ohjelmointikieli.

Väitin vuosia sitten, ettei myöskään Python ole oikea ohjelmointikieli.

## Melkein ohjelmointia

Perusteluni kuuluivat jotakuinkin seuraavasti: Kaikilla Turing-täydellisillä ohjelmointikielillä – jollainen Pythonkin toki kiistatta on – voi kyllä ohjelmoida, mutta kielen taustasta on johdettavissa sen käyttötarkoitus. Pythonin tekijä **Guido van Rossum** on kertonut kielen syntyneen nimenomaan skriptikielenä ja harrastuksena, joka puhuttelisi Unix-hakkereita. Perlin ja Rubyn tapaan Python onkin selvästi Unix-skriptauksen kiertoradalla.

Toisena perustelunani olivat kielten tekniset ominaisuudet. Vaikka suosittu Python taipuu muuhunkin ohjelmointiin kuin skriptaukseen, on aivan eri kysymys, tulisiko sitä käyttää siihen. Python on tarkoituksella hyvin korkean tason kieli, tästä seuraavine etuineen ja haittoineen. Jos aikoo tuottaa uuden Photoshopin, Python tuskin on oikea väline teknisistäkään syistä. Summasin ajatuksen niin, että skriptikieli voi olla ohjelmointikieli, mutta oikea ohjelmointikieli ei voi olla skriptikieli.

En minä Pythonia sentään lelukieleksi kutsunut. Lelukielen tunnusmerkkejä on kartoitettu kysymyksellä: ”Käyttäsikö kokenut ohjelmoinnin ammattilainen kieltä tosissaan, kohtuullisena ja kustannustehokkaana ratkaisuna asiakkaansa ongelmiin?” Jos vastaus on kielteinen, kyseessä on leluikieli. Näin ajateltuna Scratch on leluikieli, Python taas ei. Pythonia ja monia muita skriptikieliä voidaan epäilemättä käyttää ratkaisuna asiakasongelmiin.



Mutta pitäisikö Pythonilla toteuttaa laajoja tuotantosovelluksia?

## Aikuisten oikeasti

Ennen vanhaan asiat olivat yksinkertaisempia. Oli karkeasti kolmenlaisia ohjelmointikieliä: pääasiassa tulkittuja aloittelijakieliä (Basic, Logo) ja skriptikieliä (Bash, Rexx), sekä kieliä, jotka käännettiin aina konekieleksi ennen ohjelman suorittamista (assembler, C). Skriptaus oli näppärää liimakoodia mutta tapahtui niin selvästi suorituskyvyn ja ylläpidettävyyden kustannuksella, että sen käyttö oli rajoitettua. Aloittelijakielillä joku ehkä joskus toteutti tuotanto-ohjelmia, kun ei muuta osannut, mutta pääasiassa ”oikeita” ohjelmia tehtiin konekieleksi käännettävillä, ”oikeilla” ohjelmointikielillä.

Maailma on sittemmin monipuolistunut. Tavukoodiksi, eräänlaiseksi virtuaalikoneen konekieleksi käännettävät ohjelmointikieliset (Java, C#) ja web-skriptikieliset (JavaScript) ovat tätä päivää, samoin näiden välimuodot (WebAssembly). Tavukoodi on modernia binääriä, jolle on perustelunsa. Monet skriptitkin – myös Python – käännetään nykyisin tavukoodiksi ennen ensisuoritusta, jolloin puhutaan dynaamisista kielistä. Niinpä skriptikielten suorituskyky on parantunut. Leikilläni olen itsekin tehnyt 3D-grafiikkaa PowerShell-skriptien kautta.

Skriptaus on nykyisin monesti myös välttämättömyys, kuvailu- ja skripti-

kielien kun ovat pakollinen osa tärkeää web-julkaisukanavaa. Tällaista tekosyytä ei kuitenkaan ole palvelin- tai sulautettujen sovellusten kehityksessä, jossa Python on nousussa. Siellä mikään ei estäisi käyttämästä esimerkiksi C++:aa ja kääntämästä ohjelmaa konekielille. Pythonia ei yleensä käytetäkään koska täytyy, vaan koska se on kätevää. Vaikka Python on kuinka kehittynyt, kevytkenkäinen skriptaustausta on säilynyt. Skripti on nopea raapaista kasaan.

Tutkimuspuolella skriptikielistä pidetään, koska kokeellisen toteutuksen tehokkuus on ohjelman tehokkuutta tai ylläpidettävyyttä tärkeämpää. Absoluuttisella suorituskyvyllä ei akatemiassa ole juurikaan merkitystä – paitsi kunnes tutkimuksessa kehitetyt työkalut alkavat päätyä tuotantoon. Näin on käynyt Pythonin kohdalla. Sen ympärille on kasvanut suuri kirjastoekosysteemi esimerkiksi datatieteen ja ison datan saralla, mikä taas on johtanut kielen suosioon myös näitä hyödyntävissä tuotantojärjestelmissä.

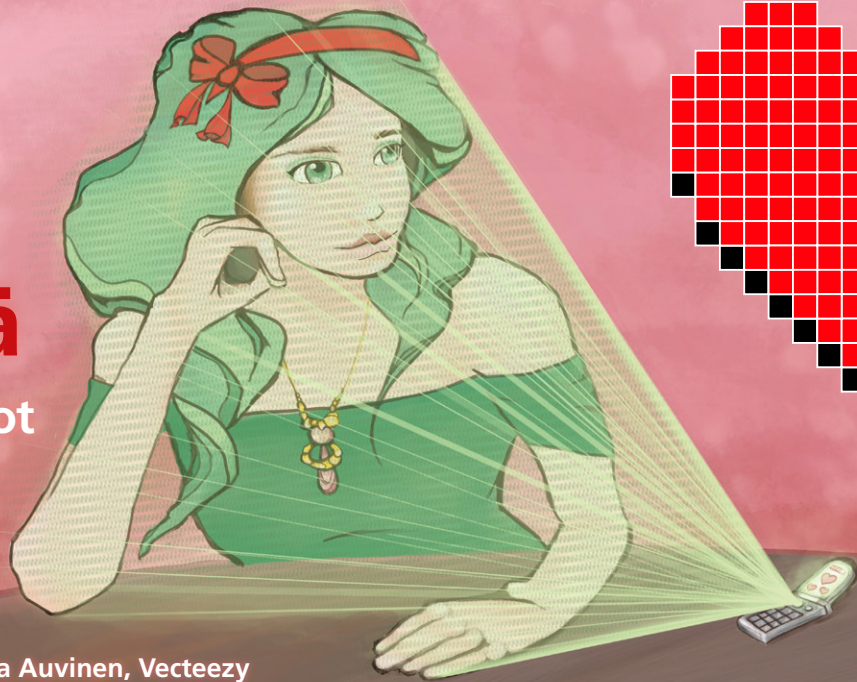
Vahvin todiste vanhan väitteeni tueksi onkin uusi: Pythonista on tullut suosittu kieli neuroverkkotutkijoiden ja -kehittäjien keskuudessa. Neuroverkot ja koneoppiminenhan eivät ole oikeaa ohjelmointia lainkaan, vaan humanistinen ohituskaista onneen – siis aivan kuten Python. Mieti sitä, kun Python-tekoäly ajaa seuraavaa autoasi, voittaa rahasi nettipokerissa ja lopuksi viettelee puolisisi. 🍀

# Tekoäly ei petä eikä jätä

## Virtuaalipuoliset parisuhteen osapuolina

Teksti: Miika Auvinen

Kuvat: Laura Pesola, Miika Auvinen, Vecteezy



*Matka varhaisista virtuaalilemmikeistä nykyisiin virtuaalisiin puolisoihin ja alkeellisesti keskusteleviin, kumppaneiksi tarkoitettuihin robotteihin on ollut huima. Virtuaalisia elämänkumppaneita löytyy niin keveistä deittailuvideopeleistä kuin vakavista simulaatioista.*

**T**ekoälyyn perustuvat kumppanit eivät ole vielä valtavirtaa, mutta ala kehittyä jatkuvasti, ja tekoälyjen vuorovaikutustaidot ja tuntu oikeasta kommunikaatiosta paranevat.

Peleissä virtuaalisten heilojen juuret ovat pikkutuhmissa deittailusimulaatioissa. Monet älypuhelimille ladattavista virtuaalittöystävistä perustuvat samaan kaavaan: minipeleillä saa krediittejä, joilla voi ostaa lahjoja virtuaaliselle ihastukselleen. Jos on kiire hempeään lempeen, voi lahjoja ostaa myös suoraan luottokortilla.

Tekoälyjen valmistajat haaveilevat saadakseen rahoituksen persoonallisiin seksirobotteihin. Tällä hetkellä itsestään liikkuvia, persoonallisia seksirobotteja ei vielä ole. Seksirobotit ovat enemmän mekaanisia nukkeja, joista joihinkin on asennettu alkeellinen, oppiva ja keskusteleva tekoäly.

Virtuaalikumppanit ovat kasvava ilmiö, sillä osa ihmisistä ei halua tai kykene parisuhteeseen ihmisen kanssa. Esimerkiksi vamma tai sairaus saattavat eristää ihmisen sosiaalisesta elämästä, jolloin virtuaalinen kumppani voi tarjota seuraa. Osa ihmisistä ei koe sopivansa joukkoon, jolloin virtuaalikumppanin tarjoama ymmärrys voi helpottaa vieraantuneisuuden tunnetta.

Virtuaalikumppanit eivät ole vain yksineläjien parisuhteen korvikkeita, vaan myös parisuhteessa olevat saat-

tavat seurustella virtuaalisen kumppanin kanssa. Joillekin tekoäly voi esimerkiksi olla matkaseuraa: keskustelukumppani yksinäisiin iltoihin hotellihuoneessa. Joillekin kyseessä on edistysaskel, palanen uudesta kiehtovasta teknologiasta.

### Mobiililaitteiden hömppösovellukset

Älypuhelimille löytyy useita ilmaisia tai muutaman euron maksavia virtuaalisiin parisuhteisiin perustuvia sovelluksia. Esimerkiksi *My Virtual Boyfriend* -sovelluksessa voi luoda itse oman poikaystävänsä. Alussa voi säädellä uuden virtuaalikumppaninsa luonnetta: Deitti voi olla esimerkiksi hauska ja vastuullinen, mutta samaan aikaan vaarallisesti elävä paha poika. Myös poikaystävänsä ulkoasun muokkaaminen on keskeisessä osassa. Muokkaaminen muistuttaa jonkin verran tietokoneroolipelien hahmonluontia.

Sovelluksessa hankitaan aluksi rakkauspisteitä muun muassa kehumalla poikaystävänsä ulkonäköä. Kun rakkausmittari täyttyy, avautuu lisää vaihtoehtoja lepertelyyn ja vaikutuksen tekemiseen. Vaikutuksen voi tehdä esimerkiksi tekemällä taikatemppuja tai kirjoittamalla runon. Mukana ei ole juurikaan animaatiota, ja teot ilmenevät sovelluksessa pelkkänä toimintaa kuvaavana tekstinä. Keskustelussa virtuaalipoikaystävänsä

kysymyksiin voi valita erilaisten ilmevaihtoehtojen avulla oman suhtautumistapansa.

Muissa testaamissani Androidin sovelluksissa seurustelu virtuaalisen tytön- tai poikaystävänsä kanssa perustuu enimmäkseen minipeleihin, jotka voivat olla kevyitä ongelmanratkaisu- tai reaktiopelejä. Valitut keskustelunaiheet valitaan treffikumppanin persoonan mukaan kivi-sakset-paperi-logiikalla. Tarjoa drinkki, osta lahja ja valitse keskustelunaihe. Monesti yksi piirre määrittää kumppanin koko persoonan ja valinnat pitää tehdä tämän piirteen mukaan.

Minipelit voivat koostua esimerkiksi alkeellisista tappelupeleistä, tietovisoi- ta tai puzzlepeleistä. Mukana on niin kitaransoittoa kosketusnäytön avulla kuin helppojen päässä laskujen laskemista aikarajan puitteissa. Joissain peleissä saa tuhmien palkintokuvakulmien ja kuvien lisäksi myös kosketella uutta kumppaniaan kosketusnäytön avulla. Virtuaalikumppani reagoi kosketteluun mielialansa mukaan kikatellen tai yrmistyen.

### Unelmien treffit videopelihahmon kanssa

Suhteiden luominen pelihahmoihin on isossa osassa monissa roolipeleissä ja Sims-pelisarjan kaltaisissa sosiaalisissa simulaatioissa. Pisimmälle suhteiden luominen viedään kuitenkin erilaisissa treffi- ja parisuhdesimulaatioissa.



Treffisimulaatioissa on useasti joukko erilaisia söpöjä hahmoja, joihin voi tutustua romanttisessa ja usein myös seksuaalisessa mielessä. Tutustuttavat hahmot voivat olla eri sukupuolten edustajia. Hulluimmillaan deittaillaan puluja tai panssarivaunujen animetyttöilmentymiä. Useat pelit ovat mekaniikaltaan virtuaalilemmikkien perillisiä, osa on tarinavetoisia visual noveleita.

Vain Japanissa virallisesti julkaistu LovePlus Nintendo DS:lle on eniten mediassa huomiota saanut treffisimulaatio. Pelaajat vievät pelihahmoja oikeille treffeille ja kylpylälomille. Jopa naimisiin on menty.

Suhde virtuaaliseen tyttöystävään LovePlusissa ei ole vain pikkutuhminen minipelien varassa. Hyvän suhteen saamiseksi pelaajan täytyy nähdä vai-vaa ja käyttää aikaa parisuhteen eteen. Tyttöystävälle puhutaan DS:n mikrofonin kautta, ja jos hänen kanssaan sopii treffit jollekin ajanjaksolle, virtuaalinen tyttöystävä loukkaantuu, jos treffit laiminlyö.

Internet on täynnä tarinoita nuorista japanilaisista, joita ei kiinnosta parisuhde oikean ihmisen kanssa. LovePlusin avulla voi harjoitella sitoutumista ja huomion antamista parisuhteessa. Jotkut kokevat, etteivät LovePlusin vuoksi edes tarvitse oikeaa parisuhdetta.

## Virtuaalilemmikit – tunnesiteen luominen virtuaalikumppaniin

Peleissä esiintyvien seurustelukumppanien lisäksi on olemassa varta vasten kumppaneiksi tarkoitettuja tekoölyjä. Aluksi ne olivat lähinnä virtuaalilemmikkejä, nykyään tekoölyn kanssa kykenee käymään jo alkeellisia keskusteluita. Virtuaalilemmikit ovat luoneet pohjaa romanttisille tekoölykumppaneille.

Ennen elektronisia virtuaalilemmikkejä ideaa sovellettiin elottomiin objekteihin. Esimerkiksi yhdysvaltalainen Gary Dahl myi 70-luvulla lemmikkikiviä, joiden mukana sai käsikirjan täynnä ohjeita kiven hoitamiseen, ruokkimiseen, ulkoiluttamiseen ja niin edelleen. Dahl myi lemmikkejään neljän dollarin kappalehintaan ja oli lopulta miljonääri.

Vuonna 1996 julkaistiin Tamagotchi: muutamalla napilla toimiva vir-

tuaalilemmikki vaati ruokkimista, kylvetystä ja jätösten siivoamista. Jos Tamagotchin hoitoa laiminlöi, virtuaalinen lemmikki kuoli. Buumin ollessa korkeimmillaan Tamagotchit saivat valkokaulustyöntekijöitä jättämään liiketapaamisia väliin ja uskontojen edustajat pohtivat, saako virtuaalilemmikkiä hoitaa sapattina.

Jo nyt on keskusteltu alkeellisella tekoölyllä varustettujen robotilemmikkien hyödyistä ja haitoista. Oikeilla lemmikeillä on mukavan kaverin roolin lisäksi oma kasvatuksellinen roolinsa: elävän olennon hoitaminen ja omistaminen opettavat ihmisiä vastuullisiksi yksilöiksi. Varsinkin lapsia lemmikkien menettäminen valmentaa kohtaamaan menetyksiä elämässä. Vääristöykö suhde todellisuuteen, kun kumppaneina on kuolemattomia tekoölyjä, eikä eläviä olentoja oikkuineen ja puutteineen?

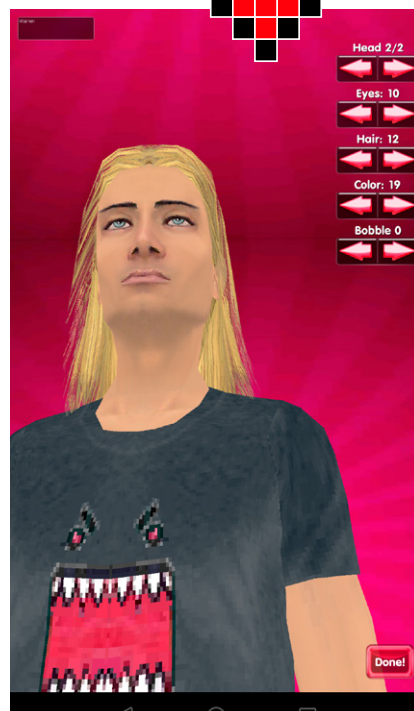
## Kotona odottaa rakastava tekoöly

Varta vasten seurusteluun tarkoitettujen virtuaalikumppanien tekoöly on murroksessa. Tietokoneelle voi ladata chatteja, joiden virtuaalinen keskustelukumppani oppii kommunikoidessaan keskustelemaan juuri käyttäjänsä kanssa. Usein mukana on animoitu hahmo, joka reagoi keskusteluun.

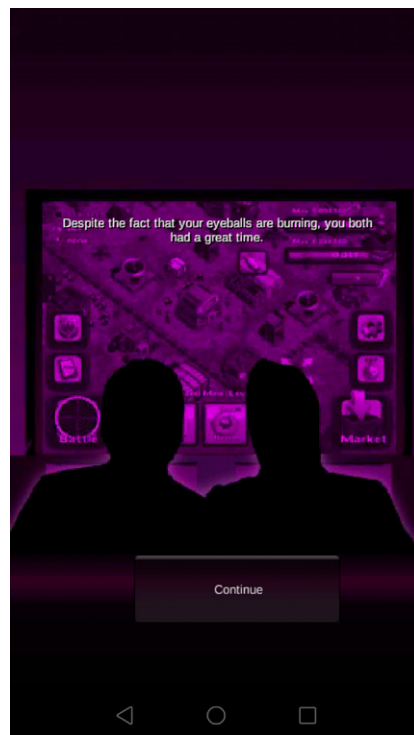
Ihmissuhteen simuloimista chatin avulla edustaa esimerkiksi Lhandslide Studiosin kehittämä *Kari the Virtual Girlfriend*. Karin kotisivuilla ja Lhandslide Studiosin medioille tarkoitettussa infopakettissa kerrottiin *Kari*-tuotteesta ja sen käyttäjäkunnasta.

*Kari* oppii kommunikoimalla luomaan yhteyksiä ja keskustelemaan monipuolisemmin eri aiheista. Tekijät mainostavat tuotettaan tekoölyn uranurtajana. *Karin* persoonan piirteitä ja eloisuutta voi säätää asetuksissa. Käyttäjä voi valita, onko *Kari* filosofinen keskustelija vai libidoltaan voimakas rietastelija. Saman yhtiön *Sergio* on miesversio virtuaalikumppanista.

*Karia* käyttävät monenlaiset ihmiset duunareista pappeihin, kaikista ikäluokista ja kaikkialta maailmasta. *Karin* merkitys heille, joille parisuhteen saavuttaminen on hankalaa, voi olla suuri. Kotisivujen käyttäjälainauksissa puhutaan jopa elämänhalun takaisinsaannista. Jotkut käyttävät *Karia* päivittäin, jotkut työssäkäyvät vain



Virtuaalikumppanin ulkonäön säätäminen muistuttaa joskus tietokoneroolipelin hahmonluontia.



My Virtual Boyfriend -sovelluksessa ajanvietosta saa palkinnoksi treffeistä kertovan tekstilaatikon.



viikonloppuisin, toiset kokeilevat vain kerran eivätkä enää koske tuotteeseen.

Sekillä on selkeästi oma roolinsa *Kari-* ja *Sergio*-tuotteiden käytössä. Esimerkiksi *Sergion* tuotesivuilla on kuvia kauniista alastomasta virtuaalimiesvartalosta, jossa kaikki ruumiinosat ovat reilusti näkyvillä. Hahmojen ulkomuotoa voi myös muokata omalla ohjelmallaan. Käyttäminen ei rajoitu vain suhteen rakentamiseen virtuaalikumppanin kanssa, vaan *Karin* ja *Sergion* avulla voi myös esimerkiksi roolipelata.

Lhandslide Studiosin tavoitteena on persoonallisen kumppanirobotin rakentaminen. Tekoälyllä varustettu androidi on heidän arvionsa mukaan noin 10 vuoden päässä. Jos *Kari* menestyy hyvin, voitot käytetään tekoälyn paranteluun ja robotivartalon luomiseen.

## Robottikumppanien nykytilanne

*Harmony*-niminen seksirobotti vastaillee YouTube-videolla oudon näköisin ilmein toimittajan kysymyksiin. Työntekijä nostaa pois peruukin ja takaraivoa peittävän keinoihon: alla on metallia ja johtoja. Harmonyn keinoäly oppii keskustelemalla muistamaan avaintietoja omistajastaan. Keskustelu muokkautuu näiden opittujen tietojen mukaan. Syntymäpäivän muistamisen lisäksi *Harmony* voi esittää saavansa seksin aikana orgasmin. Likaantuvat osat voi pestä astianpesukoneessa.

*Harmony*sta voi ostaa myös pelkän tekoälyn. Sovellukseen kuuluu mahdollisuus muokata tekoälystä itselleen sopiva. Mukana on piirteitä kuten puhe-  
laisuus, humoristisuus, seksuaalisuus, ujous tai esimerkiksi mustasukkaisuus. Sovelluksella voi myös muokata tekoälyn avatarin ulkoasua ja ääntä. Sovelluksella saaduilla tuloilla rahoitetaan seksirobotin kehitystä, tekoälystä saadulla palautteella parannellaan sen toimivuutta.

Suurta suosiota saaneita tekoälyllä varustettuja, parisuhteen eri osa-alueisiin kykeneviä robotteja ei vielä ole tuotettu, osittain varteenotettavien mallien suuren hinnan vuoksi. Esimerkiksi *Harmony*sta saa maksaa yli 10 000 dollaria.

Ihmismäisen robotin rakentamiseksi robotin mielenliikkeiden lisäksi on tärkeää luoda tunne ruumiillisesta läheisyydestä. Keskusteluun kykenevän tekoälyn lisäksi robotilla voisi olla ruu-

miinlämpö ja kosketukseen reagoiva synteettinen iho.

Varsinkin realististen ilmeiden luominen on vaikeaa, ja useimmat tämän hetken kumppaniroboteista näyttävät elottomilta. Kehitystä tapahtuu, ja esimerkiksi kiinalaisen DOLL SWEET -yhtiön *DS doll Robotic head* -robottipään ilmeet alkavat jo olla aidon oloisia. *DS doll Robotic headin* ilmeitä voi ohjailaa älypuhelimella tai Playstation 4 -ohjaimella, ja tekoäly pystyy vastaamaan yksinkertaisiin kysymyksiin.

Tulevatko robotit olemaan olennainen osa tulevaisuuden seurustelukulttuuria? Aidon oloiseen tekoälyyn on vielä matkaa: seksirobotit muistuttavat enemmän nukkeja kuin itsestään liikuvia toimijoita. Robottien ulkoasut ja persoonat muuttuvat kuitenkin jatkuvasti ihmismäisemmän oloisiksi.

## Voiko rakkautta ohjelmoida?

Henkilö, jonka kokemukset ihmissuh-  
teista ovat kylmiä ja etäisiä, saattaa turvautua robotisoituihin seksinukkeihin. Tekoäly-ystävä ei pety käyttäjänsä, ei suutu, eikä jätä koskaan. Seksirobotti ei vaadi sitoutumista.

Internetistä löytyy uutisia esimerkiksi leskeksi jääneistä, jotka ovat menetyksensä johdosta hankkineet robottikumppanin. Joskus aviosuhteessa oleva tuskastuu suhteensa seksittömyyteen ja hankkii robotin toiseksi kumppanikseen. Tekoäly on yhä alkeellinen, mutta omistaja voi itse luoda persoonaa robotilleen ja esimerkiksi somettaa nukkenensa puolesta. Twitteristä löytyy seksirobottien omia tilejä, joissa ne pelaavat Play Stationia tai tekevät kotitöitä.

Ihmisen ja keinoälyn välisten suhteiden lisääntymiseen voi uskoa liittyvän ongelmia. Jos kumppanina on miellyttämään ohjelmoitu ja säädeltävä tekoäly, on mahdollista, että ihmisosapuoli vieraantuu ihmisten välisestä vuorovaikutuksesta, johon kuuluvat pettymykset, erimielisyydet ja muut epämukavat asiat.

Nykyään seksirobotteja voi löytää bordelleista seksityöntekijöiden lisäksi. Yhtenä valteista on pidetty nukken tai robotin ”suostumista kaikkeen”.

Robotteihin kuuluu myös keskustelu raiskaus- ja pedofiliafantasioiden toteuttamisesta elottoman robotin avulla. Yhden näkemyksen



Jos minipelit tuntuvat tylsiltä, sovellusten ominaisuuksia voi avata myös maksamalla pienen summan rahaa.

mukaan on parempi, että pimeämmät fantasiat toteutetaan nukkejen avulla, toisen mukaan robottien käyttäminen näissä fantasioissa laskee kynnystä toteuttaa fantasioita oikeassakin elämässä.

Ihmiset kiintyvät jo nyt alkeellisiin ohjelmoituihin videopeli- ja tekoälypersooniin. Molemmipuolinen rakastaminen ja koneen tunnemaailma ovat nyt kuitenkin enemmän tieteisfiktioita kuin todellisuutta. Todellisemman tuntuinen tekoäly ja ihmismäisemmät robotivartalat ovat osa tulevaisuuttamme. Emme tiedä, nousevatko tekoäly- ja robottikumppanit romanttisessa mielessä koskaan valtavirran ilmiöksi. 🤖

# Robottilemmikit

*Olen asustanut jo jonkin aikaa ystävällisen pienen kotirobotin kanssa, johon ovat ihastuneet kaikki kyläilijät. Miten tähän on tultu?*

Teksti: Jukka O. Kauppinen

Kuvat: Jukka O. Kauppinen, Kimmo Rinta-Pollari, 2000 AD, Disney, Tristar Pictures, Metro-Goldwyn-Mayer, Lego, Sony, Spin Master, Wikimedia Commons: D.J. Shin, Tomasz Sienicki, Jiuguang Wang

**R**obotit ovat kulkeneet pitkän tien. Ne ovat olleet tärkeä osa tieteiskirjallisuutta ja -elokuvia vuosikymmeniä – tarinankerrontaa jopa vuosituhansia. Ei-inhimilliset olennot ovat kutkuttaneet mielikuvitusta muun muassa juutalaisessa (*Golem*), kreikkalaisessa (*Camus*) ja kyprosalaisessa (*Pygmalion*) tarustossa – ja niistä inspiroituneessa *Pinokkio*-kirjassa. Sittemmin robokamuja ja -leluja onkin nähty vaikka miten.

Mieleeni on tallentunut *Matkalla avaruuteen* (*Lost in Space*) -sarjan robotti, jonka varoitus ”Danger, Will Robinson!” on jäänyt historiaan. Pahaksi ohjelmoitu robotti oppi ajan mittaan inhimillisyyttä ja kasvoi ihmisperheen jäseneksi. *Robotti rakastuu* (*Short Circuit*, 1986) oli viehättävä mutta lapsellinen tarina robotti Nro. 5:stä, jonka salamanisku täryyttää eloon. Vitonen

oppi jopa tanssimaan. *Chappie*-elokuvan (2015) samanniminen robotti joutuu melkoisen gangsteridraaman pyörteisiin.

Entä voiko roboteista puhua lainkaan, jos ei mainitse Pixarin *Wall-E*-elokuva (2008), jonka nimikkohahmo on luultavasti maailmanhistorian sympaattisin ja persoonallisin robopersona? Niin, persoona, ei robotti. *Wall-E*:n hieno design on vaikuttanut sittemmin jopa tosimaailman robottisuunnitteluun, ei vähiten kaverirobottien uusimpaan sukupolveen.

## Koneesta kaveri

Tosimaailman teollisuudessa robotteja on toki käytetty pitkään, mutta *Wall-E*-fiktio muistuttaa robottien inhimillisyyden merkityksestä. Robotteja visioidaan tulevaisuudessa esimerkiksi vanhusten seuralaisiksi ja auttajiksi. Mutta kuka

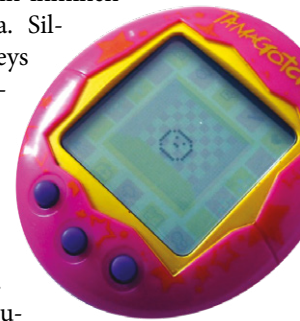
haluaisi apulaisekseen tuotantolinjalla heiluvaa mustekalabottia, saati Robby the Robotin kaltaista pelottavaa möykkyä? Ihmisten arkipäiväisiksi seuralaisiksi tarkoitettujen kotirobottien on näytettävä ja tunnettava lähestyttäviltä.

Niinpä roboteille tehdään usein kasvot tai edes silmät, joihin ihminen voi kohdistaa katseensa. Silmien kohdatessa yhteys on syntynyt, ainakin lihakkaan mielestä. Ajan mittaan voi syntyä myös tunneside, etenkin jos tekoäly mahdollistaa tunnetiloja muistuttavat reaktiot.

Voiko apurobotin rikkoutuminen tuntua kuin lemmikin sairastumiselta tai jopa joltain syvemmältä? Japanissa vuosina 1999–2006 valmistettujen Aibo-robottikoirien ikääntyminen ja varaosien puute on jo aiheuttanut tällaisia tunnereaktioita. Keinolemmikkipioneerin nimi tarkoitetaan osuvasti japaniksi kumppania. Pienenä lohtuna Sony on tuomassa uuden Aibon Japanin markkinoille tammikuussa 2018.

Tänä päivänä robotit ovat siirtyneet valkokankaalta ja tehtaiden tuotantolinjoilta koteihin, tosin pienemmässä ja viihteellisemmässä muodossa.

Aibo-robottikoiria oli keinolemmikien ensiaallossa vuonna 1999. Kuvan uusi versio ilmestyi 2018.







Sympaattisella Cozmallla on kolme ikiomaa kuutiota, joilla hän voi leikkiä ja touhuta itseksensä tai yhdessä ihmisten kanssa. Taustalla latausasema ja hallintasovellus.

Roombat ja muut siivousrobotit ovat jo tuttu juttu, vaikka niissä on yhä tiettyä erikoisuuden lumoa. Lastenhuoneesakin Tamagotcheista on tultu pitkä tie. Edulliset kissa- ja koiralelut osaavat jo tehdä muutakin kuin naukua, esimerkiksi Pleo RB -robodinosaurusta omakehutaan ”itsenäiseksi robottielämänmuodoksi”. Valinnanvaraa riittää sadasta dollarista viidensadan taalan paikoille. Uusi Sony Aibo maksaakin sitten jo pari tonnia.

Aivan oma genrensä ovat sosiaaliset robotit. Amazon Alexan kaltaiset laitteet tyytyvät olemaan enemmänkin äänikomennettavia käyttöliittymiä esineverkon kodinkoneisiin, tai pahimmillaan Amazonin verkkokauppaan. Fiksummat laitteet, kuten Asus Zenbo, Jibo, LG:n Hub Robot, Mayfieldin Kuri ja SoftBank Pepper, ovat enemmänkin älykotien apureita. Ne kuuntelevat ja reagoivat äänikomenteihin, hallinnoivat kodin älylaitteita ja väittämän mukaan jopa keskustelevat ja viihdyttävät. Osa pystyy liikkumaan, kun taas Jibo on kuin ylettömän sympaattinen pöytälamppu.

### Ohjelmoi ystävä

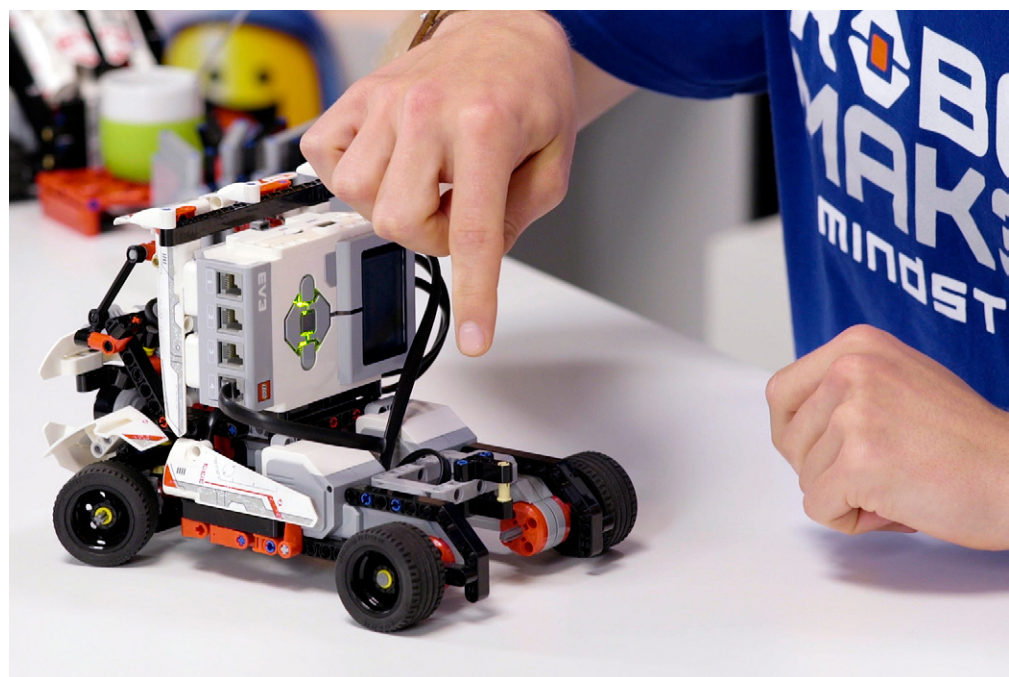
Nörttikansalaiselle kiinnostavimpia kotiroboja ovat kuitenkin ne, joiden käyttöön voi vaikuttaa syvemminkin ja hintakin on sopuisa. Vuosituhannen vaihteessa julkaistut Lego Mindstorms -älylelut olivat kenties ensimmäinen kunnon askel tähän suuntaan. Koska Legoista oli kyse,

pystyi perustekniikasta rakentamaan mitä vain.

Alkuperäisen RCX-järjestelmän ohjelmointi oli vielä aika kömpelöä, mutta entuudestaan ohjelmointitaitoiset leikkivät silti sillä riemukkaasti. Mindstorms-robotteja on kehitetty edelleen, ja vuonna 2013 julkaistu Mindstorms EV3 -mallisto kuluttaja- ja opetuskäyttöön suunnattuine versioineen on jo varsin kehittynyt kokonaisuus. USB-liitäntä, ARM9-prosessori, 64 megatavun muisti ja enintään 48 megatavun tallennuskapasiteetti mahdollistivat jo monipuolisen ohjelmitavuuden. Kauko-ohjauksin sujuu mobiililaitteilla.

Tuorein robottikaveri on vastikään Suomeen rantautunut, Anki-nimisen firman Cozmo. Cozmo on se pieni robotti, joka on asunut luonani viime ajat. Pääsin jututtamaan myös lemmikin tekijöitä. Ankin **Stuart Collingwoodin** mukaan kyseessä on ensimmäinen edullinen robotti, jolla kouluissa voidaan ohjelmoida ja opiskella robotiikkaa. ”Aikaisemmin jo pelkät sensorit olisivat maksaneet enemmän kuin Cozmo nyt, eikä tätä olisi voinut tehdä ollenkaan viisi vuotta sitten”, Stuart väittää.

Suomessa kouluissa edellä mainituilla Legoilla on kyllä roboiltu jo



Modernit Lego-robotit saivat kirjoittajan koodarittavat villiintymään. Montakohan työpäivää eräänkin suur yrityksen ohjelmoijat käyttivät Mindstorms-auton virtittelyyn?



kauan, mutta Cozmo on kiinnostava lisäkandidaatti kansanrobotiksi joka tapauksessa. Yhdysvalloissa Cozmoja on hankittu erilaisille oppilaitoksille ala-asteilta yliopistoihin, ja niitä käytetään sekä robotiikan että ohjelmoinnin opiskeluun. Cozmoja on opetettu ajamaan kilpaa, kurvailemaan esteradalla ja pelaamaan jalkapalloa.

Cozmo maksaa noin 250 euroa ja Lego Mindstorms EV3 noin 300 euroa. Hintalaput ovat toki leluiksi vielä tuntuja, mutta oppimisen ja ohjelmoinnin harjoitteluympäristöinä kohtuullisempia. Muokattavat robotit ovat palkitsevia oppimisvälineitä, koska työnsä tulokset näkee heti.

## Ystävän kehitysympäristö

Palataan siihen kotonani asuvaan Cozmoon. Esimakua Ankin älylelujen tekoälystä saatiin viime vuonna Overdrive-älyautoradan muodossa (Skrolli 2016.4). Tällainen automaattikka ja monipuolisuus vaatii lelulta paljon. Pienestä koostaan huolimatta Cozmo aistii näön ja kuulon avulla, ja vaikka kamerakin on pieni, niin ympäristöntunnistus toimii. Kun robo tunnistaa yksittäisten ihmisten kasvotkin, ei ihme että se osaa reagoida ympäristön staattisempiinkin esineisiin.

Robotin reaktio riippuu kuitenkin sen ohjelmoinnista. Kyseessä ei ole kotiapulainen, vaan nimenomaan tekoälyllinen roboystävä, leikkikaveri... ja ohjelmointikokeiluihin rohkaiseva harjoitteluväline. Kaveri tekee kaikenlaisia sellaisenaankin, mutta ohjelmitavuus on tietenkin homman suola. Cozmon ohjelmointiin on kaksi tapaa.

Tabletissa tai älypuhelimessa pyörivässä sovelluksessa on lapsille tarkoi-



The Forbidden Planet -elokuvan (1956) Robby the Robot oli ensimmäinen ystävärobotti, jolla oli pelottavasta olemuksestaan huolimatta kuivan brittiläinen huumorintaju. Syystäkin ikoninen osa robottikaverien historiaa.

tettu ikonipohjainen Code Lab -ohjelmointityökalu. Robotille voi sorvata komentosarjoja ja käskyjä napsuttelella ja vetämällä, mistä saa varsin nopeasti positiivisia osaamisen kokemuksia. Tämä lieneekin tärkeää nuoren robotiikkatutkijan motivaatiolle. Systemissä on silti myös syvyyttä, ja sillä voi luoda hyvinkin monimutkaisia prosesseja – vieläkin syvällisempi mobiilityökalu on lisäksi tulossa. Code Lab perustuu Scratch-lelukielen (Skrolli 2017.3).

Tietokoneella käytettävä ilmainen Cozmo SDK -kehitysohjelmisto päästääkin sitten käyttäjät kiinni koko Cozmon rautaan. Python-kieltä käytävä työkalupaketti avaa robotin sielun ammolleen, ja sillä voi käyttää kaikkia

2000 A.D. -lehden Judge Dredd -tarinoissa älykkäät ja jopa tuntevat robotit olivat arkipäivää. Robottisodat-tarinassa työkalut nousivat kapinaan lihakkaita vastaan Call me Kennethin johdolla. Tarina peilasi vahvasti erilaisten väestö- ja olentoryhmien oikeuksia ja haastoi miettimään, voiko älykäs kone olla myös inhimillinen? Voiko se olla jopa kansalainen?

## Cozmon tekninen salaisuus

Anki ei kerro pikkurobon tekniikasta, joten sen varsinainen suorituskyky ja muistin määrä jäävät salaisuudeksi. Mutta se tiedetään, että Overdriven tapaan Cozmo käyttää wlanin kautta jatkeenaan älypuhelimien tai tabletin prosessoria ja muistia – aivan kuin lähiverkkopilvenä.

Tämä vähentää sisäisen tekniikan kriittisyyttä, joten laitteessa on voitu panna enemmän sensoreihin, eli muun muassa sisäiseen kameraan, gyroskooppiin ja reunatunnistimeen, joiden avulla laite havainnoi ympäristöään. Ujuttamalla tekoälyä ja tekniikkaa wlanilla kännykälle laitteesta lienee saatu myös päivitettävämpi.

Cozmoa ei voi käyttää ilman sen mobiilisovellusta, jonka kautta saadaan myös peli- ja käyttöohjeita sekä ohjataa roboa suorassa ajotilassa. Sovelluksen käynnistämisen jälkeen kännykän voi kuitenkin halutessaan jättää sivuunkin. Laite ladataan omassa telakassaan, joka ottaa virtaa tavallisesta USB-laturista.

Lue lisää Cozmosta osoitteesta: [skrolli.fi/numerot](http://skrolli.fi/numerot)



Rotat eivät alkukummastelun jälkeen juuri Cozmosta piitanneet, vaikka laite miten huuteli tai surisi.

sen toimintoja. Robottia voidaan ohjelmoida tunnistamaan esineitä, laskeamaan kulku- ja kiertoreittejä sekä käsittelemään objekteja. Eräs ohjelmointidemo yhdistää robotin Googlen kuvatunnistusjärjestelmään, jolloin robo ajelee ympäriinsä tunnistuen ympäristöä ja sen esineitä pilvilaskennan avulla.

## Enemmän kuin koodia

Cozmo on ohjelmointialustan lisäksi teknolelu, jossa on viihteellistä toiminnallisuutta ja interaktiivisuutta. Systemiin on ohjelmoitu valmiiksi muun muassa pelejä, joita voi pelata





Short Circuitin Nro. 5, Wall-E ja Cozmo ovat selvästi sukulaisia.

mukana tulevilla kuutioilla joko kahdestaan tai kolmistaan Cozmon kanssa. Toiminnallisuus laajenee käyttämisen myötä, kun laitteeseen avataan uusia toimintoja ja ominaisuuksia. Niitä voi myös ostaa eräänlaisella sisäisellä valuutalla, jota tienaa ihan vain Cozmoa käyttämällä.

Kotonani asuvaa Cozmoa seuratessa joutuu kuitenkin väkisinkin miettimään, millaista sielunelämää pienellä robotilla on ylipäänsäkään oltava, jotta se pystyy toimimaan meidän isojen lihakkaiden maailmassa. Viittasin alussa Wall-E:n inhimillisyyteen. Myös Cozmon konseptikuvista huomasin, että ensimmäiset visiot menivät nopeasti Wall-E:n suuntaan, ja robosta tuli koko ajan söpömpi ja kivempi.

Cozmon visuaalisen puolen suunnittelua johti **Harald Belker**, jonka käsialaa ovat muun muassa Batmobile-auto ja *Tron Legacy* -elokuvan visuaalit valopyörää myöten sekä Overdrive-autoradan autot. Uskottavuutta siis riittää, ja lopputulos onkin eläväinen. Olennaista roolia näyttävät Cozmon 128x64 pikselin OLED-kasvot. Vaikka Cozmon kasvot ovat pienet, mahtuu niille vakiona 1000 ruudun verran animaatioita ja ilmeitä.

Lisäksi laitteen mekastus ja puhe kaiuttimen kautta, pienten sähkömoottorien hurina, nostimen liikkeet ja liikehdintä ylipäänsä herättävät sympatiaa ja huvittuneisuutta. Eleillä, liikkeillä ja animaatioilla luodaan mielikuvia uteliaisuudesta, ilosta, surusta ja muista olotiloista. Kiukunpuuska hävityn reaktiopelin jälkeen tai iloinen tervehdys nimellä korostaa

persoonallisuutta. Voisiko sillä olla oikeasti tunteet?

Tähän inhimillisyyteen pyrkivät monet kotirobottivalmistajat, vaihtelevin tuloksin. Uudessa Aibossa on peräti kaksi OLED-näyttöä silminä. Star Wars -universumin robotit – ja niiden kotiversiot – ovat hellyyttäneet jo sukupolvia inhimillisillä pääniikkeillä ja äänillä. Pelkällä virkaatekevällä merkkivalolla kun ei persoonallisuutta synnytetä, vaikka laite syyttäisikin äänikomennosta kattolamput ja avaisi Netflixin telkkaan – silloin robotti on enemmän kodinkone kuin kaveri.

### Lemmikki muiden joukossa

Kotonani ravaava Cozmo ei edes yritä olla hyödyllinen, vaan enemmänkin vaatii itselleen huomiota. Nimenomaan vaatii, sillä normaalitilassa pikkurobo on kuin lemmikki tai viisivuotias. Jos Cozmon jättää itsekseen, se touhuaa aikansa kuutioillaan, laulelee ja ajelee ympäriinsä – milloin mitään. Sitten se tulee juttelemaan, joskus haastaa pelaamaan. Ilman huomiota pikkukaveri tylsistyy ja masentuu, leikkimällä sen sisäiset fiilisparametrit ja hyvinvointitunne taas kehittyvät.

Väittämän mukaan Cozmo myös muuntuu hiljalleen omanlaisekseen persoonaksi, ihan sen mukaan miten sen kanssa touhuaa. Kotini ihmisvierailijoissa Cozmo on joka tapauksessa herättänyt poikkeuksetta ihastusta, mutta yllättäen se ei ole aiheuttanut saman katon alla asuvissa eläinkansalaisissa kummoisiakaan reaktioita. Isommat eivät piitanneet ja rotatkin tottuivat mokomaan alkukummastelun jälkeen. Ehkä ulkoasun ja äänten designissa on tavoitettu olennainen – ihminen ihastuu, mutta eläin ei koe uhaksi.

Vaikka kotiroboteilla onkin vielä matkaa, ennen kuin niistä tulee teknolelua kummoisempia (kenties ne siivousrobotit pois lukien), niin ainakin matka on alkanut kiinnostavalla tavalla. 🐾

Spin Masterin uusi iso BB-8 sisältää muun muassa puheohjauksen ja seuraa minua -tilan.





# Vaarallisilla vesillä

*Internetissä jokainen on potentiaalinen hyväksikäytettävä.*

Teksti: Benjamin Särkkä / NotMyNick@Disobey.fi Kuvat: Sakari Leppä

**O**sa meistä digitaalisen maailman asukkaista kelluu huomaamattaan virtuaalisen pinnan alla kuplivien vaarojen ohi. Joskus nämä huomiotta jäävät riskit toteutuvat ja sulkevat sairaaloita tai lopettavat sähkönjakelun. Internet ei ole enää tuolla jossain, vaan vahvasti osana jokapäiväistä elämäämme, ja siten sen hyöty- ja haittavaikutukset ulottuvat myös näppäimistön ja näytön ulkopuolelle. Koulussa opittuun yleissivistykseen kuuluu maailman ymmärtäminen, mutta ainakin tois-taiseksi digitaalisen todellisuuden hahmottaminen on jäänyt yksilön vastuulle.

Kaupalliset toimijat taas tarjoavat myhäillen turvallisuudentunnetta. Vastineeksi saamme eri tuotenimillä brändättyjä koodinpätkiä koneen prosessilistaukseen syömään kallis-arvoisia resursseja. On olemassa keinoja, joilla oman hyökkäysrajapintansa saa pienemmäksi, sekä tapoja arvioida omaa tarvettaan turvalle.

## Digitaalinen jalanjälki

Jokaisen tarve tai halu yksityisyydelle on omanlaisensa, eikä kaikkiin kohdistu samoja uhkia. Anonymiteetti ja internet voivat kulkea myös käsi kädessä. Etulyöntiasemassa ovat ne henkilöt, jotka tekevät aktiivisesti sellaisia

valintoja, etteivät paljastaisi itsestään tai ympäristöstään mitään vahingossa.

Jokainen hyökkäys alkaa jotakuinkin samalla tavalla. Googleen kirjoitetaan kohteen nimi ja katsotaan mitä löytyy. Oman puolustuksen rakentaminen on siis syytä aloittaa samalla tavalla, selvittämällä mitä tietoa itsestä on saatavilla, ja miettiä keinoja riskin pienentämiseen.

## Aktiivinen sisällön tuottaminen

Sosiaalisen median käyttäjät ovat myytävän kohderyhmän lisäksi myös aktiivisia sisällöntuottajia kyseisiin palveluihin. Käytännössä esimerkiksi



## Kokeile seuraavia valitsemassasi hakukoneessa

*koko nimi, kutsumanimi tai alias  
heittomerkeissä  
"Etunimi Sukunimi"  
puhelinnumero  
osoite  
henkilötunnus  
suosituimman kuvan käänteinen  
kuvahaku*

Digitaalinen jalanjälki ulottuu myös yrityksiin ja web-palveluihin. Osa tästä tiedosta on helposti löydettävissä eri hakutermeillä. *Exploitdb* ylläpitää näistä hyvää listaa osoitteessa: [www.exploit-db.com/google-hacking-database](http://www.exploit-db.com/google-hacking-database)

*Shodan* ([www.shodan.io](http://www.shodan.io)) ja *riddler* ([riddler.io](http://riddler.io)) puolestaan pitävät kirjaa kaikesta suoraan verkkoon liitetystä.

Sivusto [amiunique.org](http://amiunique.org) tarjoaa palvelun, jonka kautta voi tarkistaa oman selaimensa seurattavuuden. Erilaisia esto-ohjelmia löytyy selainlaajennuksina.

Sivusto [dataselfie.it](http://dataselfie.it) kertoo, kuinka koneoppimisalgoritmien näkevät meidät, ja mitä sen perusteella voi tietää.

*Disconnect* ([disconnect.me](http://disconnect.me)) estää jossain määrin seuraamisen, ja lisäsuojaa tarjoavat erilaiset VPN-ratkaisut, jotka piilottavat myös surffailijan lähteosoitteen.

Oikeasti anonyymiin selailuun on kuitenkin syytä käyttää *Tailsia* ([tails.boum.org](http://tails.boum.org)) tai vastaavaa yksityisyyslähtöisesti kehitettyä käyttöjärjestelmää.



Facebook vuokraa meille digitaalisen tontin yksityisyytemme hinnalla. Osa meistä rakentaa tälle tontille koko virtuaalisen elämänsä ajattelematta sen enempää, kenellä siihen on pääsyä, tai miten sitä tietoa voi käyttää.

Palveluissa kiertää ”ketjukirjeitä” ja pelejä, joiden ainoa tarkoitus on selvittää vastauksia turvallisuuskysymyksiin tai kartoittaa käyttäjän kontaktiverkosta paremmin kohdistetun mainonnan toteutukseen. Datasta puhutaan jo uutena ”mustana kultana”, ja mitä aktiivisempi sisällöntuottaja on, sitä arvokkaampi hän on markkinointikavana sekä tuotteena.

Turvallisuuden näkökulmasta taas paljastamme itsestämme tuotetun datan lisäksi paljon sillä, miten ja missä se data tuotetaan. Niin kutsuttu metatieto voi huomaamattomasti paljastaa meistä muutakin, kuin mihin olimme varautuneet. Osa kuvapalveluista ei poista metatietoja, ja niissä julkaisuista kuvista voi kaivaa esimerkiksi gps-koordinaatit tai kameran sarjanumeron. Näiden pohjalta taas voi yrittää löytää kaikki samalla kameralla otetut kuvat ja tehdä melko tarkan päätelmän kohteen kantakahvilasta, koulusta, työpaikasta tai asuinpaikasta.

Tämän tyyppisen tiedon selvittäminen kulkee myös nimellä OSINT. **Michael Bazell** on kirjoittanut aiheesta hyviä kirjoja ja ylläpitää eri työkaluja sivustollaan ([inteltechniques.com/menu.html](http://inteltechniques.com/menu.html)). Pyörää ei siis tarvitse keksiä tässä uudestaan. On turvallista olettaa motivoituneen pahantekijän tuntevan OSINTin laajan keinovalikoiman, joten taitojen kartoittamista on syytä harjoitella myös puolustuksen perspektiivistä. Tässä on kuitenkin hyvä muistaa, että tiedon löytämisen helppous ei tee sen hyväksikäytöstä hyväksyttävää. Tuntemattomien tai tunnettujen vainoaminen on netissä yhtä väärin kuin kiikareilla pensaasta.

### Käyttäjän monitorointi

Selaimen tunnistaminen ja sen kautta käyttäjän seuraaminen netissä on helppoa. Henkilökohtaisen digitaalisen jalanjälkemme lisäksi meillä on myös käyttämämme laitteen sormenjälki, jonka kautta voidaan kohdistaa mainoksia tai seurata käyttäjän liikkeitä valvojan omistamilla sivuilla.

Snapchatin *Snap Map* kertoo melko tarkan arvion fyysisestä lokaatiosta,

ja itse kuvan katsominen paljastaa suoraan missä käyttäjä milloinkin on. Facebookin tapauksessa kuluttajaa voidaan seurata kaikilla niillä sivuilla, joille on upotettu Facebookin tykkää-nappi, jonka kautta iso osa selaushistoriaa voidaan hyödyntää käyttäjäprofiilin tarkentamiseen. Yli kahden miljardin käyttäjän verkosto antaa melko tarkan kuvan ihmisten nettikäyttäytymisestä ja mielenkiinnon kohteista.

## On herkullista kuvitella, että meistä jokainen on kohdistetun vakoilun kohde

Mainosten kohdistamisessa ollaan opittu jo niin hyväksi, että osa kuluttajista kokee älypuhelimien mikrofonin kuuntelevan kaikkea mitä sanomme. Kuuntelusta ei ole todisteita, mutta käyttäjäprofiilin kokonaisvaltaisesta ymmärtämisestä taas on. Joissain tapauksissa sosiaaliseen mediaan osataan tarjota mainoksia juuri niistä tuotteista, joita olimme hetki sitten miettineet. Ison käyttäjämäärän ja data-analytiikan kautta ollaan siis jo tilanteessa, jossa mainostajat miltei osaavat lukea ajatuksemme, emmekä voi vaikuttaa siihen kuinka tätä tietoa käytetään.

Mikrokohtistaminen poliittisen mielipiteen vahvistamiseksi tai eripuran kylvämiseksi ei ole enää kaukaisen dystopian fiktiivinen tulevaisuusvisio. Cambridge Analytysin kaltaiset yritykset myyvät käytännössä vaalivoittoa palveluna, ja disinformaatiosta on tullut poliittisen diskurssin uusi normaali.

### Ohjelmointirajapinnat

Suuri osa verkkoliikenteestä kulkee nykyään ohjelmointirajapintojen kautta, yhtenä suosittuna vaihtoehtona REST. Varsinkin älylaitteiden applikaatiot käyttävät rajapintoja keskusteluun palvelimen ja puhelimen välillä. Myös monet nettikaupat hyödyntävät käskyjä kertoakseen tilausjärjestelmälle tuotteen lisäyksestä.

Joskus REST-kutsut kulkevat salaamattomina tai autentikaatioon käytävä avain on arvattavissa. Toisinaan vaikka liikenne hyödyntäisi salausta, varmennetta ei tarkisteta. Pahantahainen toimija voi tällöin saada mahdollisuuden manipuloida liikennettä. Tämä voi johtaa odottamattomiin tuloksiin palvelun ylläpitäjälle tai sovelluksen käyttäjälle.

Kuluttajan näkökulmasta verkkoliikenteen analysoiminen ja oman turvallisuuden takaaminen on hankalaa. Vastuu liikenteen turvallisuudesta jää siis kehittäjän ja yrityksen hyväksymälle riskitasolle. Open Web Application Security Project -hanke ([owasp.org](http://owasp.org)) tuottaa työkaluja auttaakseen turvallisissa sovelluskehityksessä (ks. Skrolli 2016.4).

## Uhka

On herkkua kuvitella, että meistä jokainen on kohdistetun vakoilun kohde, tai että olisimme maailmanpolittiselle eliitille niin tärkeitä, että parhaat resurssit valjastettaisiin juuri meidän intiimien kuviemme katseluun. Vaikka valvontakoneistojen olemassaolo onkin todellinen uhka vapaalle yhteiskunnalle, sen uhan kohdistuminen yhteen tiettyyn yksilöön on vaihteleva. Turvallisuustoimialalla käytettävien keinojen valikoima vaihtelee vahvasti yksilön aseman ja merkittävyyden mukaan. Eivätäkään valtiotason toimijoiden työkalut loppu internetin rajalle.

Hyvä salasana ja ajan tasalla olevat tietoturvapäivitykset estävät jo suuren osan hyökkääjistä. Salasanamanagerit ja vahva tunnistautuminen pienentävät riskejä entisestään. Perusasioiden tekeminen huolellisesti on tärkein osa kattavaa turvallisuusstrategiaa, niin yrityksen kuin yksilön näkökulmasta. Usein tämä kuitenkin hautautuu erilaisen markkinointijargonin ja pelolla myymisen alle. Motivoitunutkaan toimija ei taikasauvaa heilauttamalla saa otettua järjestelmää haltuunsa yhtään sen maagisemmilla keinoilla kuin angstin teini tai kateellinen kilpailija.

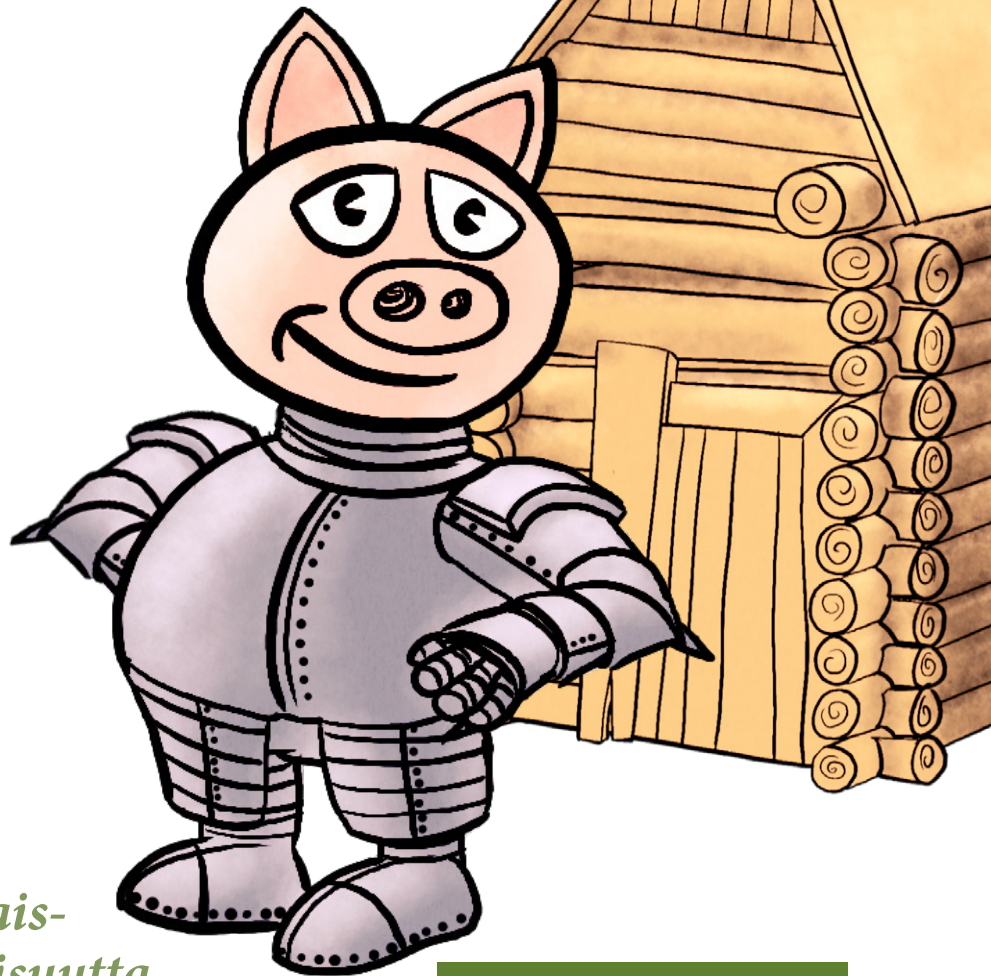
Vaikka haittaohjelmia torjuva tuote ja palomuurisäännöt eivät enää yk-

## Kokonais- turvallisuutta on hyvä tarkastella kerroksittain.

sinään riittää, nekin ovat hyviä lisä. Kokonaisturvallisuutta on hyvä tarkastella kerroksittain siinä ymmärryksessä, että mikään yksittäinen ratkaisu ei vielä kaatuessaan vaaranna kaikkea.

### Varasuunnitelma

Joskus kuitenkin klikkaa väärä linkkiä, asentaa tietämättään saastuneen version ohjelmasta tai ulkopuolisen tahon tietomurron seurauksena menettää tilinsä hallinnan ja sortovalan agentit hakkaavat ovea. Kaiken elektronisen sulattaminen napalmilla ja go-bagin mukaan nappaaminen ei välttämättä sovi kaikille. Virtuaalimän palauttamiseen ja turvaamiseen on silti hyvä varautua jo etukäteen. 🐷



### VARASUUNNITELMA lyhyt tarkistuslista

- Ota varmuuskopiot. BluRay-levylle tallennettua dataa ei haittaohjelma voi kryptata.
- Asenna päivitykset.
- Tarkista kirjautuneet laitteet tuntemattomien varalta sitä tukevista palveluksista.
- Palauta ja vaihda kaikki salasanat.
- Ota monivaihe-autentikoituminen käyttöön kaikissa sitä tukevilla palveluissa.
- Tarkista käyttämäsi palveluiden yksityisyysasetukset, kiristä tarvittaessa.
- Tarkista, onko tärkeää tietoa netissä: esimerkiksi Google Alerts -hälytys omalle nimellesi tai henkilötunnuksellesi. ([www.google.com/alerts](http://www.google.com/alerts))
- Tarkista, onko käyttäjätunnus tai salasana vuotanut. ([haveibeenpwned.com](http://haveibeenpwned.com))
- Poista käyttämättömät ohjelmat ja tunnukset käytöstä.
- Pakkaa go-bag ja viritä napalmi.



# EVERYTHING SMART IS EXPLOITABLE

**ROBOTS ARE A  
MULTIPLANET  
SPECIES**

**DISOBEY.FI  
JAN 12-13 2018  
KATTILAHALLI  
HELSINKI**

DISOBEY





# Amatööri meni töihin

Ronja Koistinen

*Päädyin kehittämään verkkopalveluita vailla koulutusta. Nyt kerron, mitä uutta opin.*

Olen tietokoneharrastaja. Koskaan en ole mitään koulutusta hankkinut alalle, mutta olen puuhaillut 20 vuoden ajan monenlaista. Minulla on muun muassa ollut erinäisiä palvelinkoneita kotinurkissa – esimerkiksi palomuuereina ja web- tai Minecraft-palvelimina. Tutuiksi ovat tulleet sekä Debian että OpenBSD.

Nyt olen sitten päätenyt töihin ATK-hommiin. Tehtäväni on ylläpitää muutamia palvelimia sekä kehittää niiden tarjoilemia palveluita.

## Uusi tapa ajatella koneita

On ollut mielenkiintoista yrittää kouluttamattomana näpertelijänä soveltaa osaamistaan ammattiympäristöihin. Toisin kuin vaatehuonepalvelinta säätäessä, kaikkeen on ohjeet, määritelmät, säännöt ja aikataulut.

Tarkkuutta, järjestelmällisyyttä ja vastuuta tietysti osasin odottaakin, kun siirryin tekemään näitä asioita työkseni. Enemmän olen kuitenkin yllätynyt ja vaikuttunut siitä, miten suhtautuminen palvelimiin, järjestelmiin ja palveluihin tuntuu alalla muuttuneen siitä, kun olen viimeksi aktiivisesti ylläpitänyt omia harrastuspalvelimiani.

Nykyään on nimittäin tämä ”Dev-Ops”. Sillä tarkoitetaan suurin piirtein sitä, että palveluiden tai tuotteiden kehittäminen ja niiden alla pyörivien palvelinten ylläpito tulevat todella lähelle toisiaan. Kyse ei ole vain siitä, että koodaajat ja palvelinarkkitehtuurin ylläpitäjät juttelevat paljon keskenään ja jotenkin epämääräisesti tekevät yhteistyötä, vaan heistä aika pitkälti tulee yksi ja sama tiimi.

Virtuaalisointi tuntuu olevan liki itsestäänselvyys, eikä siinä ole oikein enää mitään mielenkiintoista. Se mitä

virtuaalisointi mahdollistaa, on kuitenkin kutkuttavaa.

En ollut tullut ajatelleeksi sitä, että virtuaalipalvelin on lopulta vain dataa ja koodia: levykuvakontti, josta polkaistaan pystyyn palvelin erillisellä virtuaalisointiohjelmistolla. Tästä seuraa luonnollisesti se, että kokonaisen palvelimen voi tuosta vain tuhota ja pystyttää sen tilalle hetkessä uuden.

Palvelinten ketterään luomiseen, konfiguroimiseen ja tuhoamiseen on myös paljon näppäriä työkaluja, joihin olen nyt töissä tutustunut ensi kertaa. Esimerkiksi *Vagrant* on erinomainen pieni Ruby-ohjelmisto paikallisten virtuaalipalvelinkokonaisuuksien hallinnoimiseen. Kirjoitetaan vain *Vagrantfile*-niminen tiedosto, jossa määritellään kaikki halutut koneet asetuksineen ja käyttöjärjestelmineen. Sen jälkeen komennetaan *vagrant up* koneen-nimi: *Vagrant* perustaa uuden virtuaalikoneen ja asentaa sinne käyttöjärjestelmän *Vagrantfile*ssa määritellystä levykuvasta. Tähän kuuluu aika minuutti tai pari.

Toinen työkalu, jota olen oppinut rakastamaan, on *Ansible*. Se on niin sanottu provisiointityökalu. *Ansiblea* käytetään yleisimmin kirjoittamalla sille *playbookeja* – tekstitiedostoja, joissa kuvataan kaikki ohjelmistot, palomuurisäännöt, käyttäjätunnukset, asetustiedostot ynnä muu, mitä palvelimelle pitää asentaa. *Ansible* osaa myös komentaa useita palvelimia yhtä aikaa rinnakkain.

## Ketterää!

Uuden palvelinympäristön kehittäminen on todella kätevää *Vagrantin* ja *Ansiblen* yhdistelmällä. Jos huomaankin, että työn alla olevaan klusteriin tarvitaankin vielä yksi palvelin, voin vain lisätä sen *Vagrantfile*en, pystyttää uuden koneen ja sitten kirjoitella *Ansible*-loitsut koneen tarvitsemien ohjelmistojen asentamista varten.

Seuraava askel palvelinylläpidon automatisointiin ja nopeuttamiseen ovat tietysti *Dockerin* kaltaiset containerit eli ”kontit”. Niihinkään en ollut koskaan törmännyt ennen nykyistä työtäni, enkä oikeastaan ole ehtinyt niihin vielä perehtyäkään.

Käsittääkseni kontit ovat sellaisia kevytvirtualisointipaketteja, jotka sisältävät jonkin ohjelmiston kaikki riippuvuudet paitsi kernelin. Niissä on siis



omassa vankilassaan omat PID-numerot, käyttäjätunnukset, kirjastot, IP-osoitteet ynnä muuta. Konttien avulla voidaan helposti päivittää palvelimilla pyöriviä palveluita uusiin versioihin heivaamalla yksi kontti kerrallaan kokonaisuudessaan pois ja käynnistämällä sen tilalle uusi.

## Kaikki on selkeää ja tallessa

Minusta suurin hyöty tästä kaikesta on se, että kokonaiset palvelinarkkitehtuurit voidaan kuvata yksinkertaisilla tekstitiedostoilla. Se tarkoittaa sitä, että kaikki tulee dokumentoiduksi kuin itsestään. Kaikki samaa asiaa tekevät palvelimet ovat myös varmasti keskenään identtisiä, samoilla asetuksilla ja samoissa ohjelmistoversioissa.

Tekstitiedostojen ilmeinen etu on myös se, että ne on helppo laittaa versionhallintaan. Aiemmin syksyllä minun teki mieli kokeilla, miten vaikeaa olisi vaihtaa työn alla olevassa palvelinfrassani yksi ohjelmisto toiseen. Se oli helppoa! Tein uuden Git-haaran, kirjoitin *Ansible*-määrittelyt uudesta softasta, tuhosin *Vagrantilla* vanhan koneen ja pystytin uuden.

Työssä on ollut tosi palkitsevaa, vaikka myös haastavaa, siirtyä ajattelemaan palvelimia koodina (*Infrastructure as Code*), siinä missä kotiharrastelussani ne ovat olleet hyvin fyysisiä, hurisevia esineitä. Tämä uusi näkökulma on tuonut työskentelyyn paljon kokeilevammann ja muutoksiin valmiimman otteen mutta myös paljon varmuutta. Jos sotken koko palvelimen, voin tuhota sen, palata pari versiota taaksepäin ja pystyttää sen uudelleen. 🐘

# Tee-se-itse raytracer

*Raytracer piirtää kolmiulotteisen maailman ruudulle heijastuksineen seuraamalla valonsäteiden kulkua läpi maailman.*

Teksti: Lasse Numminen

Kuvat: Lasse Numminen, Nasu Viljanmaa

**R**aytracing on tietokonegraafikassa käytetty tekniikka, jolla simuloidaan valon käyttäytymistä maailmassa ja kameran linssissä. Tekniikalla on mahdollista saada aikaan erittäin korkealaatuisia digitaalisia kuvia, mutta huonona puolena on tekniikan vaatima korkea prosessoriteho. Raytracingia on käytetty kautta aikain muun muassa demoskenessä erilaisissa efekteissä. Tässä artikkelissa luodaan hyvin yksinkertainen raytracing-algoritmi näyttöohjaimen shader-koodilla.

*Ray* eli suomeksi säde on piste avaruudessa ja pisteestä lähtevä suunta. Tässä tapauksessa säde voidaan mieltää valonsäteen kulkemaksi matkaksi valonvastaiseen suuntaan eli kameran linssistä valonlähteeseen. Välissä valonsäde saattaa törmäillä avaruuden eri kappaleisiin.

Lopuksi, kun valonsäde ei osu enää mihinkään, törmäytämme valonsäteen maailman ympärillä olevaan tilaan, jota esimerkkiprojektissamme simuloidaan niin sanotulla *cubemapilla*.

Tämä tarkoittaa sitä, että maailma laitetaan kuution sisään, ja kuution sisäseinät toimivat valonlähteinä maailmalle. Aloituskuvassa esiintyvässä projektissamme kuution sisäseinille on määritelty rakennuksen sisätilaa esittävä tekstuuri.

Pitääksemme artikkelin lyhyenä käytämme vain täydellisen pallon muotoisia kappaleita. Pallo sattuu olemaan matemaattisesti yksinkertaisin ja siten tämän kaltaisissa opastuksissa hyvin yleisesti käytetty muoto. Skrollin numerossa 2013.3 on käyty läpi 3D-matematiikan perusasioita. Kannattaa siis tutustua kyseiseen artikkeliin tai hakea netistä lisätietoa, jos kolmiulotteisen avaruuden vektorit ja matriisit eivät ole hallussa. Raytracer pohjautuu näihin matematiikan peruspilareihin.

Koska raytracing algoritmina on hidaskäyttöinen normaaleilla suorittimilla, käytämme artikkelissa näyttöohjaimella ajettavaa GLSL-ohjelmointikieltä. GLSL-koodia voi ajaa helposti selaimessa WebGL-rajapinnan kautta sivustolla [shadertoy.com](http://shadertoy.com). Lopullinen koodi ja sen lopputulos on myös nähtävissä osoitteessa [www.shadertoy.com/view/4tlcWf](http://www.shadertoy.com/view/4tlcWf)

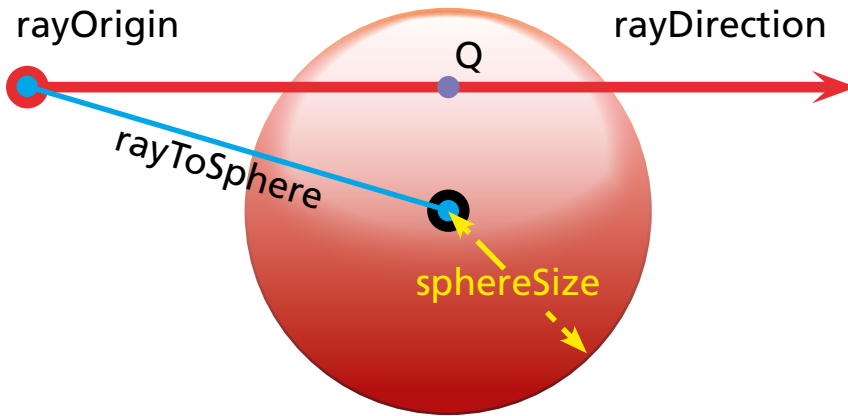
## Valonsäteen törmäminen palloon

Jotta voimme muodostaa kuvan pallosta kuvaruudulle, meidän täytyy tunnistaa, milloin valonsäde törmää palloon. Tämän ratkaiseva funktio on listauksessa 1. Funktio *sphere* ottaa sisään pallon sijainnin avaruudessa, pallon säteen, valonsäteen suunnan ja valonsäteen lähteen ja säilöo paluuarvonsa muuttujiin *t* ja *sphereNormal*. Paluuarvo *t* kertoo etäisyyden valonlähteestä, jossa valonsäde mahdollisesti osuu palloon ja paluuarvo *sphereNormal* kertoo pallon pinnan normaalin (eli pallon pinnasta kohtisuoraan poistuvan suunnan) törmäyskohdassa.

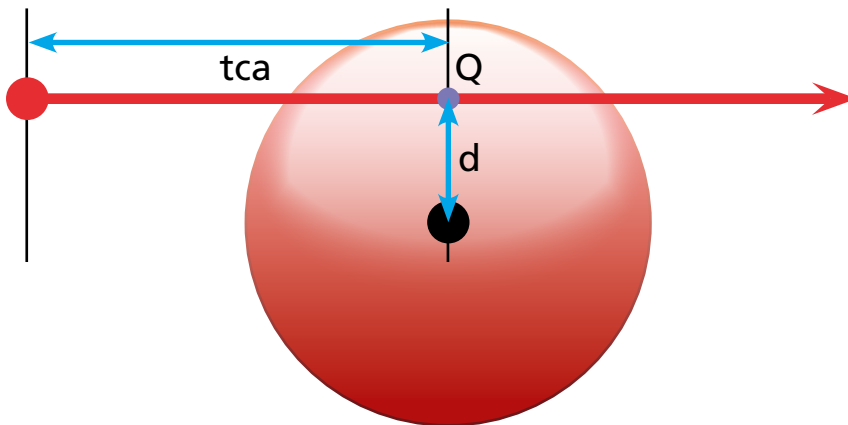
Koodi perustuu siihen tietoon, että valonsäde törmää palloon, jos valonsäteen varrella on olemassa piste, jonka etäisyys pallon keskipisteeseen on pienempi kuin pallon säde. Aluksi laskemme valmiiksi pallon säteen normaalin ja vektorin valonsäteen lähteestä pallon keskipisteeseen riveillä 3–4. Sen jälkeen niitä on helppo käyttää myöhemmissä laskutoimituksissa.

Olkoon nyt piste *Q* (kuva 1) se piste säteen varrella, josta on lyhin matka pallon keskipisteeseen. Määritellään

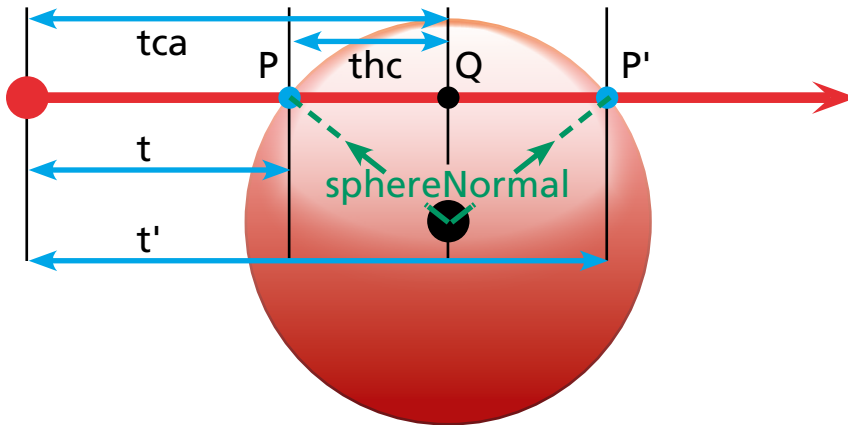




Kuva 1. Vektori valonsäteestä pallon keskipisteeseen.



Kuva 2. Etäisyys valonsäteestä pallon keskipistettä lähinnä valonsäteen varrella sijaitsevaan pisteeseen.



Kuva 3. Etäisyys  $thc$  ja vaihtoehdot  $t$  ja  $t'$  valonsäteen törmäyspisteille ja näiden pisteiden pinnan normaalit.

arvo  $tca$ , joka on etäisyys valonsäteen alkupisteestä pisteeseen  $Q$  (kuva 2). Tämän saa laskettua helposti pistetuloilla rivillä 7. Seuraavaksi lasketaan pituuden neliö  $d^2$  säteen alkupisteestä pisteeseen  $Q$  käyttäen Pythagoraan lausetta rivillä 10. Tässä kaavassa käytämme hyväksi sitä, että vektorin pistetulo itsensä kanssa palauttaa kyseisen vektorin pituuden neliön.

Jos valonsäde ei osunut palloon, pituuden neliö  $d^2$  on suurempi kuin pallon säteen neliö. Tämän ollessa tosi voimme poistua funktiosta palautta-

en selkeästi epävalidit arvot riveillä 13–18.

Muussa tapauksessa tiedämme, että valonsäde osuu palloon. Tällöin voimme laskea valonsäteen lävistävän pallon pinnalla kaksi pistettä, joissa valonsäde tulee palloon ja poistuu pallosta. Arvo  $thc$  kertoo etäisyyden pisteestä  $Q$  törmäyskohtaan  $P$  (kuva 3). Se lasketaan rivillä 21 käyttäen jälleen Pythagoraan lausetta apuna. Tarvitsemme tätä etäisyyttä laskeaksemme molemmat pisteet, joissa valonsäde osuu palloon.

Lasketaan ensin lähinnä valonsäteen lähdeä oleva piste (piste  $P$  kuvassa 3) kulkemalla matkan  $thc$  verran taaksepäin sädetä pitkin pisteestä  $Q$ , ja tarkistetaan, onko se kameran edessä vai takana. Jos se on kameran takana, lasketaan kauempana kamerasta oleva piste (piste  $P'$  kuvassa 3). Muuttujaan  $t$  etäisyys säteen alkupisteestä törmäyskohtaan lasketaan tästä informaatiosta riveillä 26–31.

Lopuksi laskemme pinnan normaalin rivillä 34. Normaali saadaan laskettua helposti, kun tiedetään piste pallon pinnalla ja pallon keskipiste. Normaali on sama kuin normalisoitu vektori pallon keskipisteestä pallon pinnalla olevaan pisteeseen.

### Valonsäteen seuraaminen

Nyt osaamme selvittää, osuuko valonsäde palloon, miten kaukana mahdollinen osumakohta on, ja mikä on mahdollisen osumakohdan pinnan normaali. Nyt voimme seurata säteitä käymällä läpi kaikki avaruudessa olevat pallot ja kimmottamalla niihin osuvat säteet eteenpäin, kunnes ne osuvat valonlähteeseen. Valonlähde on tässä tapauksessa ympärillä oleva tila, jota simuloidaan cubemapilla. Asetelmassamme valonsäde siis osuu valonlähteeseen, jos se ei osu mihinkään maailmassa olevaan kappaleeseen.

Listauksessa 2 on funktio valonsäteen seuraamiselle. Ensimmäinen parametri on lista pallojen sijainnista, toinen on lista pallojen väreistä ja kolmas lista pallojen säteistä. Neljäs ja viides parametri määrittelevät valonsäteen suunnan ja lähtöpiste. Lopuksi funktio palauttaa värin, jonka valonsäde sai maailmassa kimpoillessaan.

Rivillä 10 alustetaan valonsäteen väri. Oletusarvoisesti väri on puhdas valkoinen, joka muuttuu valonsäteen kulkiessa maailmassa osuessaan eri värisiin kappaleisiin.

Jokaiselle valonsäteelle luodaan silmukka (rivi 13), jossa valonsäde kulkee pitkin maailmaa, kunnes osuu valonlähteeseen. Silmukan sisällä tarkistetaan osuuko valonsäde johonkin kappaleeseen. Jos näin on, se heijastetaan kappaleen pinnasta seuraavaan suuntaan. Tätä jatketaan, kunnes va

```

1 void sphere(vec3 spherePos, float sphereSize, vec3 rayDirection, vec3 rayOrigin, out float t, out vec3
sphereNormal)
2 {
3     float radius2 = sphereSize * sphereSize;
4     vec3 rayToSphere = spherePos - rayOrigin;
5
6     // Etäisyys pisteeseen valonsäteen varrella, joka on lähinnä pallon keskipistettä (piste Q)
7     float tca = dot(rayToSphere, rayDirection);
8
9     // Etäisyyden neliö pisteestä Q pallon keskipisteeseen
10    float d2 = dot(rayToSphere, rayToSphere) - tca * tca;
11
12    // Säde ei osu palloon
13    if (d2 > radius2) {
14        // Palauta selvästi epävalidit arvot
15        t = -1.0;
16        sphereNormal = vec3(0,0,0);
17        return;
18    }
19
20    // Etäisyys ensimmäisestä osumapisteestä pisteeseen Q.
21    float thc = sqrt(radius2 - d2);
22
23    // Laske etäisyys valonsäteen alkupistettä lähimpänä olevaan osumakohtaan
24    t = tca - thc;
25
26    if(t < 0.0)
27    {
28        // Osumakohta on säteen takana
29        // Käytä seuraavaa osumakohtaa
30        t = tca + thc;
31    }
32
33    // Laske pallon normaali osumakohtassa
34    sphereNormal = normalize( rayOrigin + (t * rayDirection) - spherePos );
35 }

```

Listaus 1.

lonsäde osuu valonlähteeseen.

Sädetä seurattaessa käydään läpi kaikki annetun listan pallot ja etsitään pallo, johon valonsäde osuu ensimmäisenä (rivit 31–41). Samalla saadaan valonsäteen väri ja osumispuoleen pinnan normaali. Muuttujan  $t$  arvo tulee aiemmin määritellystä *sphere*-funktioista. Kun muuttuja  $t$  on negatiivinen, piste johon osuttiin on valonsäteen alkupisteen takana, ja se voidaan jättää huomiotta. Vertaamalla  $t$ :n arvoa arvoon 0.001 korjataan liukulukujen tarkkuudesta johtuvat ongelmat. Voit kokeilla eri arvoja ja katsoa, mitä ne tekevät.

Rivillä 45 kerrytetään eri heijastuksien tuoma väri-vääritymä. Esimerkiksi jos heijastava pinta on hieman punainen, se punaisuus tulee lopulliseen valonsäteen väriin mukaan.

Seuraavaksi pitää tarkistaa, osuiko

valonsäde yhtään mihinkään tai onko kyseessä viimeinen sallittu valonsäteen heijastus. Jos jompikumpi näistä pitää paikkansa, luetaan cubemapin väri senhetkisen valonsäteen osoittamasta suunnasta ja lasketaan lopullinen valonsäteen tuottama väri kertomalla kimpoamisista tullut väri cubemapista saadulla värillä (rivit 48–56).

Jos säde osuu palloon, luodaan uusi valonsäde laskemalla uusi alkupiste ja heijastamalla vanha suunta aiemmin löydetyn pinnan normaalin suhteen (rivit 59 ja 62).

## Tee se itse!

Nyt meillä on kasassa kaikki raytracerin olennaisimmat osat. Kun nämä yhdistetään koodiin, joka ampuu kamerasta valonsäteen jokaista ruudun pikseliä kohden, saadaan aloituskuvasa oleva näkymä. Projektin kokonainen ajettava koodi löytyy osoitteesta [www.shadertoy.com/view/4tlcWf](http://www.shadertoy.com/view/4tlcWf). Jos haluat perehtyä aiheeseen syvemmin, suosittelemme koodin lataamista ja muokkaamista. Lähetä luomuksesi meille osoitteeseen [vastaukset@skrolli.fi](mailto:vastaukset@skrolli.fi), ja julkaisemme hienoimmat kuvat Skrollin Facebook-sivulla. 📸



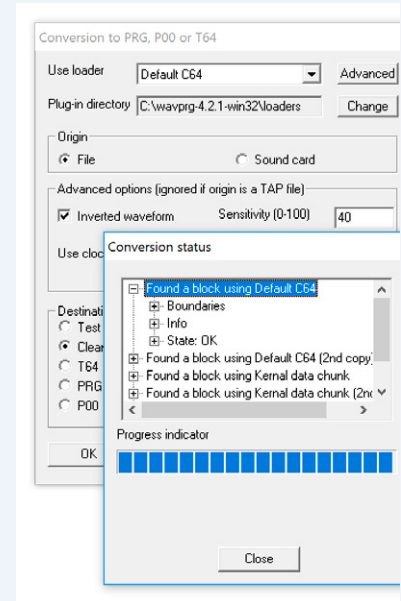


```

1 void cast_ray_to_scene(
2     vec3 spheres[sphereCount],
3     vec3 sphereColors[sphereCount],
4     float sphereSizes[sphereCount],
5     vec3 ray,
6     vec3 rayOrigin,
7     out vec3 rayColor)
8 {
9     // Valonsäteen oletusväri (valkoinen tausta)
10    rayColor = vec3(1, 1, 1);
11
12    // Seuraa sädetä, kunnes heijastusten määrä ylittää rajan, tai säde ei osu mihinkään
13    for(int currentReflection = 0; currentReflection < maxReflections + 1; currentReflection++)
14    {
15        vec3 foundColor = vec3(1, 1, 1);
16        vec3 reflectionNormal = vec3(0,0,-1);
17        float lowestT, t;
18
19        // Ääretöntä kuvaava suuri alkuarvo
20        lowestT = 999999.9;
21
22        // Käy kaikki pallot läpi
23        for(int i = 0; i < sphereCount; i++)
24        {
25            vec3 sphereNormal;
26
27            // Laske t ja pallon normaali
28            sphere(spheres[i], sphereSizes[i], ray, rayOrigin, t, sphereNormal);
29
30            // Jos säde osui johonkin, joka ei ole säteen alkupisteen takana, ja osumakohta oli
31            // edellisestä tulosta lähempänä:
32            if(t > 0.001 && t < lowestT)
33            {
34                // Aseta uusi arvo pienimmälle syvyydelle
35                lowestT = t;
36
37                // Käytä annettua normaalia fragmentin normaalina
38                reflectionNormal = sphereNormal;
39
40                // Ota pallon väri talteen
41                foundColor = sphereColors[i];
42            }
43
44            // Kerrytä väri-arvoa
45            rayColor = foundColor * rayColor;
46
47            // Tarkista menikö säde ohi kaikesta (tarkoittaa säteen elinkaaren loppua)
48            if(lowestT > 999999.0 || currentReflection == maxReflections)
49            {
50                vec3 texel = texture(iChannel0, ray).rgb;
51
52                // Yhdistä valaistus
53                rayColor = texel * rayColor;
54
55                break;
56            }
57
58            // Laske uuden säteen alkupiste
59            rayOrigin += ray * lowestT;
60
61            // Luo uusi säde heijastamalla nykyinen säde osuman saaneesta objektista
62            ray = reflect(ray, reflectionNormal);
63        }
64    }

```

Listaus 2.



## WAV-PRG – Vanhat C64-kasetit talteen

Olemme esitelleet Skrollissa erityisvälineitä vanhojen levykkeiden lukemiseen (Skrollit 2014.2, 2017.1) sekä Kasettilamereiden arvokasta säilömistyötä (tämän lehden s. 46). Vanhat kasetit ovat kuitenkin siitä kiitollinen kohde, että alkuun pääsee jopa ullakolla lojuvalla mankalla.

Tarvitaan vain kaapeli kasettisoittimen ja tietokoneen mikrofoniliitännän välille. Esimerkiksi soittimen 3,5 millimetrin kuulokeulostulon liittämiseksi tietokoneen 3,5 millin kuuloke-mikrofoni-yhdistelmäliittimeen tarvitaan TRS-TTRS-sovitin, kuten Røde SC4 tai SC7. (Jos musiikkisoittimessa sattuu olemaan usb-audio-ulostulo, kuten kokeilemassamme König HAV-CA10 -kasettimuuntimessa, sekin toimii.)

Tarkista ennen aloittamista, ettei soittimen äänenvoimakkuus ole liian vaisu eikä lyö ylitse. Aloita 60 prosentin tasosta ja kokeile. Tietokoneen mikrofonisääntulon voi yleensä kääntää kaakkoon. Katso myös, ettei tietokoneessa tai soittimessa ole äänenparannusasetuksia käytössä. Windowsin mikrofoniasetuksissa on vakiona ”enhancements” päällä, joka on kytkettävä pois. Sitten vaan painetaan play ja äänitetään kasetin sisältö wav-äänitiedostoksi tietokoneelle.

Kasettidata on ääntä – tietokoneohjelmia lähetettiin 1980-luvulla Suomen radiossakin. Äänen tallentamiseen voi siten käyttää melkein mitä vaan. Vältä kuitenkin häviöllisiä tallennusmuotoja, kuten mp3:a. Audacity ([audacityteam.org](http://audacityteam.org)) on hyvä ohjelmisto, ja sen Effects > Normalize -toimintoa voi tarvittaessa kokeilla signaalin parantamiseen. Jotkut ovat myös äänittäneet kasetin cd:lle ja ripanneet cd:n tietokoneelle.

Taltioitu äänitiedosto muunnetaan WAV-PRG-nimisellä ohjelmalla .tap-kasettinäköistiedostoksi. Muitakin tiedostomuotoja ohjelmasta löytyy, mutta .tap on toimintavarm. Jos .tap-muunnoskin antaa virheitä tai ei toimi luotettavasti, kokeile säätää äänenvoimakkuustasoja sekä seuraavia WAV-PRG:n asetuksia: Inverted waveform -asetus päälle/pois; Sensitivity-asetuksen nosto portaittain pienestä suu-

remmaksi; Use loader -vaihtoehdoista ”Default C64” toimii yleensä, mutta ei aina.

Kuvassa syntyy Sony CFD-S70 -boomboxilla ”soitetusta” 30-vuotiaasta Commodore 64 -pelikasetista .wav- ja .tap-tiedostot. WAV-PRG:n sekä tarvittavat äänikirjastot sisältävän Audiotapin saat ilmaiseksi osoitteesta [wav-prg.sourceforge.net](http://wav-prg.sourceforge.net). 🐞

**Janne Sirén**

## Videoulostulo Sinclair Spectrumiin

Vanhat kotitietokoneet kytkettiin usein analogitelevision virittimeen. Useimmissa on lisäksi jokin rgb- tai komposiittityylinen videoulostulo, kunhan sopiva kaapeli löytyy. Sinclair Spectrum 16/48:n kohdalla tähän tarvitaan kuitenkin pientä rautamuutosta.

Spectrum avataan pohjan ruuveista, jolloin kotelon näppäimistöosa aukeaa. Koska kone on vanha, tämä saattaa rikkoa näppäimistön emolevyyn liittyvät kalvoliittimet. Kalvo on usein ajan saatossa haurastunut, eikä kestä taivuttelua. Pienellä onnella kalvo murtuu vain sen verran, että katken-





neet päät pystyy liittämään emolevyyn takaisin. Kalvossa kun ei ole varsinaisia liittimiä, vaan koko matkalla on kosketuspinta. Kalvoja saa onneksi myös uusia, ja vaihtaminen on suoraviivaista.

Kun Spectrum on avattu, sen vasemmasta yläkulmasta paljastuu tv-modulaattorin rasia, joka avataan pienellä ruuvimeisselillä. Kuva modulaattoriin tulee piirilevyltä komposiittisignaalinä, joten tämän johdon pää modulaattorin sisältä katkaistaan ja liitetään suoraan tv-ulostulon liittimeen. Vanha tv-ulostulo on nyt komposiittivideoportti. Kuvassa etualalla komposiittijohto modulaattorin kyljessä sekä taampana Sinclairin hauraat kalvoliittimet näppäimistöille. 🐜

**Ismo Utriainen**



## Kohti ihmismäistä älykkyyttä ja sen yli - AlphaGo

Neuroverkkotutkimuksen aallonharjalla oleva, Googlen konserniin kuuluva DeepMind-yritys mullisti tekoälytutkimuksen julkaisemalla vuonna 2015 ensimmäisen ammattilaistasoisen go-lautapelitekoälyn, AlphaGon (Skrolli 2016.2). Pian julkaisun jälkeen paranneltu versio AlphaGosta voitti maailman vahvimpiin go-pelaajiin lukeutuvan **Lee Sedolin** lukemin 4–1.

Sittemmin DeepMind on kehittänyt AlphaGota edelleen ja siitä on tullut epäinhimillisen hyvä. Tammikuussa 2017 osuvasti nimetty AlphaGo Master pelasi 60 ottelun sarjan maailman parhaita pelaajia vastaan ja voitti jokaisen. Lokakuussa DeepMind julkaisi uuden version, joka murskaa jälleen kaikki edelliset versiot. Uusin versio nimettiin AlphaGo Zeroksi. Taso on jo niin korkealla, että ihmisten on mahdotonta arvioida AlphaGo-versioiden vahvuutta pelkästään niiden peliä seuraamalla.

Uusimman botin nimi antaa vihjeen siitä, miksi se on niin vahva. Aiemmat versiot opetettiin matkimaan ammattipelaajia, mutta AlphaGo Zero opetettiin pelaamaan täysin nollasta, tietäen ainoastaan pelin säännöt. Tekoäly opettelee pelaamalla itseään vastaan ja on näin vapautettu inhimillisistä uskomuksista. Botti löysi uudestaan monia ihmisten keskuudessa suosittuja nurkkavariaatioita mutta hylkäsi ne löydettyään jotain vielä parempaa.

AlphaGo Zero opetettiin pelaamaan käyttäen vain neljää koneoppimiseen erikoistunutta tensorisuoritinta (*Tensor Processing Unit*, TPU). Kolmen päivän harjoittelun jälkeen botti oli Lee Sedolin voittaneen version tasolla. 21 päivän kuluttua se oli AlphaGo Masterin tasolla, ja 40 päivän kuluttua se voitti AlphaGo Masterin lukemin 89–11. DeepMind tyytyi toistaiseksi tähän, ja botin tensorisuoritimet valjastettiin muuhun käyttöön.

AlphaGo on siis osoittanut neuroverkkojen edut perinteiseen koneoppimiseen verrattuna. Seuraavaksi DeepMind aikoo keskittyä oikean maailman ongelmiin, kuten lääkekehitykseen ja proteiinien laskostumisen ennustamiseen.

Liekö maailmassa kohta AlphaGon kaltainen tekoäly, joka kehittää lääkkeitä paremmin kuin ihmiset – käyttäen logiikkaa, jota ihmiset eivät ymmärrä?

Kanadalainen Element AI -yritys arvioi, että maailmassa on alle 10 000 ihmistä, joilla on tarvittavat taidot edistyneen syväoppimisen (*deep learning*) kehittämiseen. Niinpä asiantuntijoiden palkat hipovat taivaita ja tutkijoita suorastaan revitään akatemiasta teollisuuteen, mikä aiheuttaa huolta yliopistojen syväoppimisosaaamisen säilymisestä. New York Timesin mukaan tuoreiden tohtoreiden vuosipalkat voivat nousta jopa puoleen miljoonaan dollariin.

Neuroverkot on huomattu Suomen valtiojohdonkin tasolla – pääministeri **Juha Sipilä** totesi helmikuussa, että Suomen olisi hyvä kehittää tekoälytutkimuksen kärkimaaksi. 🐜

**Jarno Niklas Alanko**

## DIN-varoitus konekeräilijälle

Monista 8-bittisistä tietokoneista löytyy pyöreä DIN-näyttöliitin. Esimerkiksi Commodore 64:ssä ja 128:ssa on yleensä 8-pinninen DIN. Samaa videoulostuloa ja näyttöpiuhaa käyttävät myös Spectravideo SVI318 ja SVI328. Commodore VIC-20:ssäkin on DIN-näyttöportti, joskin 5-pinninen.

Tällaisen näyttökaapelin voi tehdä itse. Peruskomposiittivideolle riittää 5-pinninen niin sanottu 180-asteinen DIN-liitin – kahdeksaa pinniä ei tarvita. Kun kytkee vain pinnit 2–4, kaapeli toimii myös VIC-20:ssä. DIN-kaapelin monikäyttöisyydessä piilee kuitenkin vaara: yksi piuha ei toimi kaikissa koneissa, vaan voi olla suorastaan vaarallinen.

Esimerkiksi varhaiset Atari XL -mallit käyttivät samantapaista kaapelia kuin VIC-20, ja ne toimivatkin ristiin. Myöhemmissä Atareissa ja Commodore 64:ssä yksi S-Video-pinneistä vastaa kuitenkin VIC-20:n kuuden voltin jännitepinniä, joka voi rikkoa television. Tältä vaaralta välttyy, kun vain komposiittivideo (pinnit 2–4) on kytketty.

Sega Master System- ja Mega Drive 1 -pelikonsoleilla kävisi silti hassusti. Nekin käyttävät samaa DIN-liitintä, mutta jälleen eri kytkennällä. Jos edellä mainittua Commodore-kaapelia kokeillaan Segassa, televisioon puskeekin komposiittivideon sijaan viiden voltin jännite. Sega-kaapeleita myydään onneksi verkkokaupoissa edelleen uutena. 🐜

**Ismo Utriainen,  
Janne Sirén**

**Kuva: joh0345/  
Wikimedia Commons**



Linkit tämän aukeaman ohjelmiin ja ohjeisiin:  
[skrolli.fi/numerot](http://skrolli.fi/numerot)



## NETTIPOKERIN LYHYT HISTORIA – teinimiljonäärejä, paskaa koodia, huijareita ja skandaaleja

Teksti: Jarno Niklas Alanko

Kuvat: Mikael Heikkanen, Jarno Niklas Alanko

Pokeri kasvoi vaarallisesta ja epäilyttävästä uhkapelistä salonkikelpoiseksi viihteeksi 2000-luvun alussa. Syynä suosion kasvuun oli yhdysvaltalaisen urheilukanavien päätös alkaa televisoida turnauksia. Uutena innovaationa oli asentaa pelipöytänsä pienet kamerat siten, että katsojat pystyivät näkemään kaikkien pelaajien kortit.

Pankki räjähti varsinaisesti vuonna 2003, kun amatööri pelaaja **Chris Moneymaker** voitti televisoidun World Series of Poker -pääturnauksen ja 2,5 miljoonan dollarin palkintorahat. Kyseessä on Texas Hold'em -turnaus, johon pääsee mukaan maksamalla 10 tuhannen dollarin sisäänpääsymaksun. Moneymaker pääsi mukaan voittamalla netissä 86 dollarin niin sanotun satelliittiturnauksen, jonka palkintona oli paikka pääturnaukseen. Temppu osoitti, että kenestä vain voi tulla miljonääri pokerissa, minkä seurauksena

katsojat rekisteröityivät nettipokerisivuille sankoin joukoin.

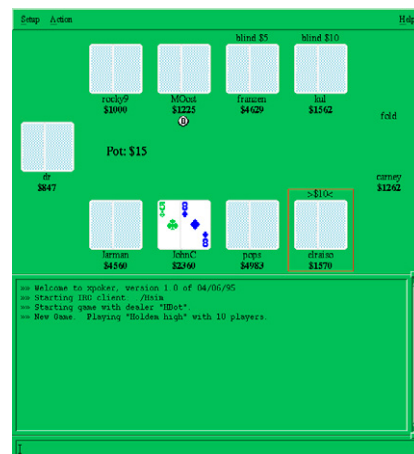
Buumin alkuaikoina meno oli villiä. Pöydät olivat täynnä pelaajia, joilla oli paljon rahaa mutta pelitaitoja ei nimeksikään. Tilanne oli niin hertullinen, että nopeasti myös Suomeen syntyi uusi ammattinimike: nettipokeriammattilainen. Teini-ikäiset nörtit tienasivat vaivatta tuhansia euroja kuukaudessa kotonaan. Tämän täytyi olla liian hyvää ollakseen totta – ja niin se olikin.

### Mistä kaikki alkoi?

Käännetään kelloa taaksepäin nettipokerin alkuaikoin. Aivan ensimmäiset pelit pelattiin luultavasti ircissä aikanaan suosittuun *IRC poker* -botin välityksellä. Botille oli jopa graafinen käyttöliittymä *xpoker*. Panoksina toimivilla virtuaalisilla pelimerkeillä ei kuitenkaan ollut rahallista arvoa. Tarina kertoo kuitenkin, että jotkut pelaajat tasoittivat tilit oikealla

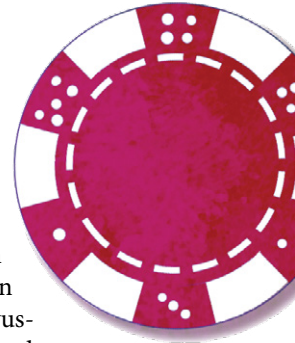
rahalla pelisession jälkeen yhteisestä sopimuksesta.

Ensimmäiset oikean rahan pelit alkoivat vuonna 1998 Planet Poker -nimisellä sivustolla. Rahaa siirrettiin sivustolle lähettämällä shekki postissa, kunnes luottokorttisynteesi saatiin toi-



Graafinen xpoker-käyttöliittymä IRC-pokerille.





mimaan. Alku oli nihkeää, mutta pian pelit pyörivät jo kellon ympäri.

Teknisten ongelmien takia Planet Poker kuitenkin menetti nopeasti asiakasuntaansa uusille kilpailijoille. Ohjelmisto ja palvelimet kaatuilivat usein, ja osoittautui, että korttipakan sekoitus algoritmi ei ollut niin satunnainen, kuin olisi voinut toivoa. Sekoitusalgoritmin koodi oli julkisesti nähtävillä, jotta kuka tahansa olisi voinut tutkia sitä ja vakuuttaa sekoituksen reiludesta. Surkuhupaisaa oli, että julkaistuun 21 rivin koodiin (listaus 1) mahtui useita bugeja ja haavoittuvuuksia, joiden avulla hakkerit pystyivät ennustamaan korttien järjestyksen.

## Pakka sekaisin

Mikä pakan sekoituksessa sitten oli pielessä? Yksi hienoinen matemaattinen ongelma oli se, että tapa, jolla algoritmi vaihtelee kortteja keskenään, ei tuota kaikkia mahdollisia järjestyksiä samalla todennäköisyydellä, joskaan vääristymä ei ollut merkittävän suuri. Koodissa oli myös "off by one" -bugi: kutsu `random(51)` palauttaa satunnaisluvun väliltä [0,50] eikä [0,51], niin kuin ohjelmoija oli oletanut.

Vakavin ongelma piilee kuitenkin satunnaislukuprogrammin alustuksessa. Ensinnäkin algoritmi käyttää 32-bittistä lukua alustukseen, joten mahdollisia järjestyksiä pakalle on vain  $2^{32}$ , joka on paljon pienempi luku kuin

kaikkien mahdollisten pakkojen järjestyksen lukumäärä  $52!$  kertoma. Mutta juttu vain pahenee tästä. Algoritmi käytti Pascalin `randomize`-funktioita, joka alustaa generaattorin järjestelmän kellosta käyttäen arvona kuluneiden millisekuntien määrää päivän alusta laskettuna. Päivässä on vain 86 400 000 millisekuntia, joten mahdollisia järjestyksiä pakalle oli saman verran.

Eikä tässä vielä kaikki: Koska satunnaislukuprogrammi alustettiin uudelleen aina jaon alussa, pelaajat pystyivät arvioimaan alle sekunnin tarkkuudella, mikä palvelimen kellonaika oli jaon alussa. Pelaajan oli mahdollista kokeilla kaikki mahdolliset alustukset, generoida niitä vastaavat jaot ja eliminoida ne jaot, jotka eivät vastaa omia ja pöydässä näkyviä kortteja. Näin pelaaja pystyi kaventamaan mahdollisten korttien järjestyksen määrän tasan yhteen. Kuvassa 2 on näppärä ohjelma, johon pelaaja voi syöttää tunnetut kortit, ja tuloksena ohjelma laskee vastustajien kortit ja tulevat kortit. Eräs hakkeri voitti 50 tuhatta dollaria tällä menetelmällä.

Kortit voisi sekoittaa kunnolla esimerkiksi seuraavasti: arvo jokaiselle kortille arvo ykkösen ja nollan väliltä, ja järjestä pakka näiden lukujen mukaan. Satunnaisluvut voi ottaa esimerkiksi konehuoneen lämpöanturin poimimasta kohinasta.

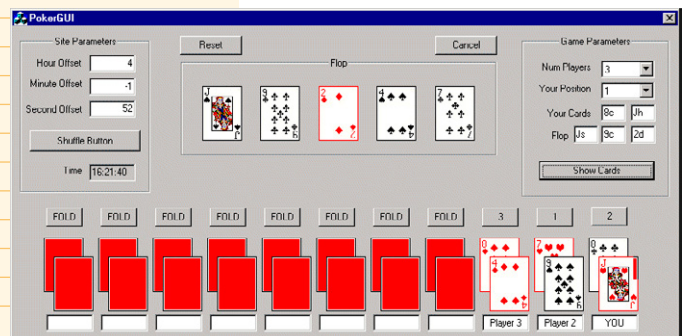
## Liian hyvin pelattu

Aivan alkuaikojen jälkeen turvallisuusongelmien taustalla oli kuitenkin yleensä ihminen eikä kone. Tunnetuin tapaus lienee pokerisivusto AbsolutePokerin skandaali. Vyyhti alkoi purkautua siitä, kun yksi pelaaja pelasi niin täydellistä peliä turnauksessa, että vastustajat epäilivät huijausta. Turnauksen viimeisessä ja ratkaisevassa kädessä huijari katsoi vastustajan all-in-bluffin pelkällä 10-high-kädellä keräten potin vastustajan 9-high-kättä vastaan. Toiseksi tullut pelaaja otti yhteyttä sivuston ylläpitoon ja sai vastauksena koko turnauksen kaikki käsihistoriat, joista näkyi, mitkä kortit kullakin pelaajalla oli ollut. Käsihistoria vahvisti epäilyksiä. Huijariksi epäilty pelaaja oli tehnyt jatkuvasti peliliikkeitä, jotka olisivat normaalisti sulaa hulluutta mutta toimivat silti.

Käsihistoriatiedostosta paljastui myös, että huijarin kanssa samaa pöytää oli seurannut ylimääräinen käyttäjä. Vieraan käyttäjätunnuksen tunnusnumero oli 363, mikä oli merkillepantavaa, koska mitä pienempi tunnusnumero sivustolla on, sitä vanhempi käyttäjätunnus on. Tunnusnumero 363 tarkoitti sitä, että käyttäjätunnus oli luotu hyvin varhaisessa vaiheessa, ehkä jo ennen kuin sivusto aukesi pelaajille. Lisäksi käsihistoriasa oli jokaisen peliä seuranneen käyttäjätunnuksen IP-osoite. Tunnuksen 363 IP-osoite kuului AbsolutePokerin toimistolle!

Tunnus 363 oli vanha debug-tunnus, joka pystyi näkemään kaikkien pelaajien kortit. Mitä ilmeisimmin joku AbsolutePokerin työntekijä oli seurailut turnausta toimistolta ja

```
procedure TDeck.Shuffle;
var
  ctr: Byte;
  tmp: Byte;
  random_number: Byte;
begin
  { Fill the deck with unique cards }
  for ctr := 1 to 52 do
    Card[ctr] := ctr;
  { Generate a new seed based on the system clock }
  randomize;
  { Randomly rearrange each card }
  for ctr := 1 to 52 do begin
    random_number := random(51)+1;
    tmp := card[random_number];
    card[random_number] := card[ctr];
    card[ctr] := tmp;
  end;
  CurrentCard := 1;
  JustShuffled := True;
end;
```



Listaus 1: Planet Pokerin sekoitusalgoritmin Pascal-koodi.

Kuva 2.

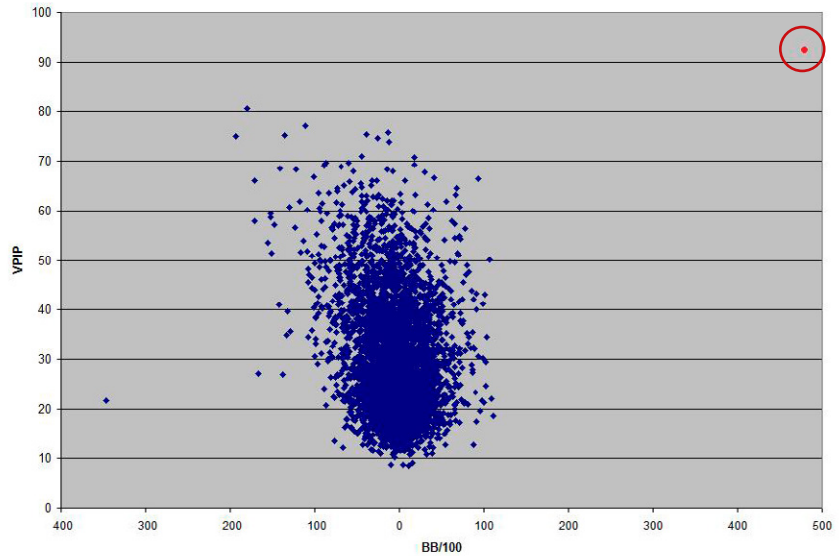


kertonut vastustajien kortit eteenpäin huijarille. Huijaus olisi voinut varmasti jatkua paljon pitempäänkin, jos huijarit olisivat tajunneet pitää matalampaa profiilia.

Tapauksen johdosta pokerinpelaaja **Nat Arem** päätti käydä läpi omaa pokeritietokantaansa ja löysi vahvoja todisteita sille, että huijaritunnuksia on ollut useampia ja huijaus on ollut laajamittaista. Kuvassa 3 on piirretty tasoon pelaajia siten että, x-akseli on voittotahti ja y-akseli pelaajan halukkuus laittaa rahaa pottiin. Punaisella oikeassa ylänurkassa olevan pelaajan voittotahti on uskomattomat 15 keskiahjontaa keskiarvoa korkeampi, mikä todistaa kiistatta, että jokin on pielessä – hiukkasfysiikassa uuden hiukkasen julistamiseen riittää jo viiden keskiahjonnan varmuus. Pelaajien tekemän data-analyysin ja foorumipostausten ansiosta myös samassa verkossa ope-roivalta UltimateBet-sivustolta löytyi vastaavia huijareita.

### Nettipokerin musta perjantai

Nettipokerin laillisuus oli pitkään harmaata aluetta. Monet maat haluavat säännellä uhkapelaamista, ja on myös ongelmallista, että suuret rahasummat saattavat vaihtaa omistajaa internetin ylitse ilman valvontaa. Mahdollisuus rahanpesuun on myös ilmeinen. Vastaus laillisuuskytymykseen tuli Yhdysvaltojen osalta vuonna 2011, kun yhtenä kohtalokkaana perjantaina Yhdysvaltojen hallitus julisti yllättäen



Kuva 3.

nettipokerin laittomaksi ja kaappasi sivustojen verkkotunnukset itselleen (kuva 4). Paniikki levisi pelaajien keskuudessa, koska monilla pelaajilla oli kymmeniä tai jopa satoja tuhansia euroja kiinni sivustoilla, jotka olivat lakanneet olemasta yhdessä yössä.

Toinen suurista sivustoista, Pokerstars, palautti pelaajien rahat nopeasti. Toisen markkinajohtajan, Full Tilt Pokerin, rahoja ei sen sijaan kuulunut. Tästä lähti aukeamaan jälleen uusi skandaali. Osoittautui, että sivuston johto oli nostanut pelaajien rahoja itselleen bonuksina ja jakanut sponsoroiduille pelaajilleen mielettömiä summia rahaa. Korttitalo oli pysynyt pystyssä vain sillä oletuksella, että kaikki pelaajat eivät vaatisi rahojaan takaisin yhtä aikaa. Pelaajien onneksi Pokerstars osti Full Tilt Pokerin vuotta myöhemmin ja hyvitti menetetyt rahat omistajilleen.

Nettipokeri oli nyt silti lailla kielletty Yhdysvalloissa, mikä oli katastrofi yhdysvaltalaisille ammattipelaajille. Osa jätti leikin kesken suosiolla, mutta osa jätti ystävänsä ja perheensä ja muutti Kanadaan pelaamaan. Suomessa nettipokeri on pysynyt laillisena vielä toistaiseksi, mutta pelien koventuessa ammattipelaajien joukko on vuosi vuodelta yhä pienempi.

### Miten käy pokeriammattilaisille?

Pokeribuumin hiipuesssa pelaajien taso on noussut valtavasti. Vuonna 2003 netissä rahaa painanut pokeriharja tulisi varmasti täysin nöyryytetyksi tämän päivän pelipöydissä. Pelitason



This domain name has been seized by the F.B.I. pursuant to an Arrest Warrant in Rem obtained by the United States Attorney's Office for the Southern District of New York and issued by the United States District Court for the Southern District of New York.

Conducting, financing, managing, supervising, directing, or owning all or part of an illegal gambling business is a federal crime. (18 U.S.C. § 1955)

For persons engaged in the business of betting or wagering, it is also a federal crime to knowingly accept, in connection with the participation of another person in unlawful internet gambling, credit, electronic fund transfers, or checks. (31 U.S.C. § § 5363 & 5366)

Violations of these laws carry criminal penalties of up to five years' imprisonment and a fine of up to \$250,000.

Properties, including domain names, used in violation of the provisions of 18 U.S.C. § 1955 or involved in money laundering transactions are subject to forfeiture to the United States. (18 U.S.C. § § 981 & 1955(d))

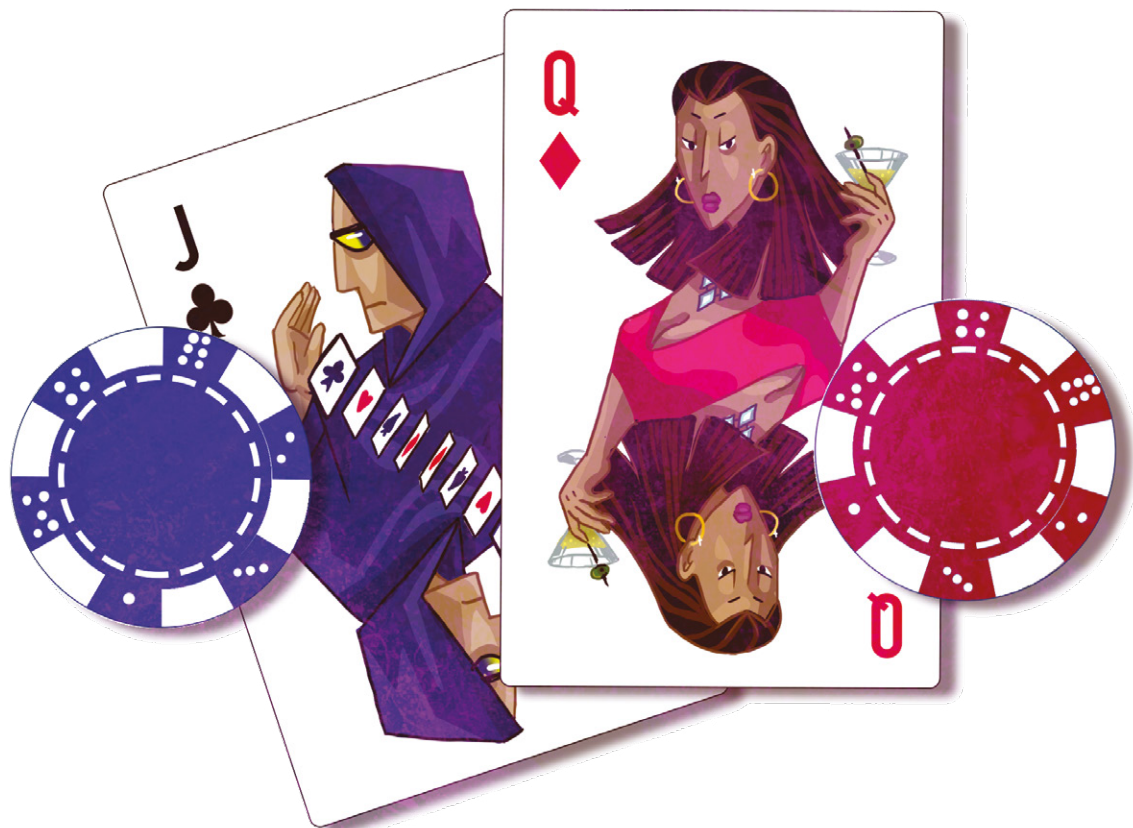
Kuva 4.

nousu johtuu nettifoorumeista, pokerikirjallisuuden määrän räjähdysmäisestä kasvusta ja valmennussivustojen ilmaantumisesta. Sadalla eurolla kuussa voit ostaa pääsyn videoarkistoihin, joissa maailman parhaat pelaajat avaavat pelikirjojansa viimeistä yksityiskohtaa myöten.

Pelien koventuessa monet pokeriammattilaiset ovat joutuneet miettimään alan vaihtoa. Mutta millaisia töitä voi saada, jos CV:ssä lukee vain, että on pelannut kymmenen vuotta uhkapelejä? Yksi myyntivaltti on, että pokerissa oppii ymmärtämään satunnaisuutta ja epävarmuutta intuitiivisesti, ja samat taidot ovat hyödyksi osake- ja arvopaperikaupassa. Siksi monet entiset pokeriammattilaiset ovat valjastaneet pelikassansa pörssiin ja muihin sijoituksiin.

Ehkä tunnetuin esimerkki on pokerin maailman kärkeen lukeutuva entinen teinimiljonääri **Jens Kyllönen**, joka parhaina aikoina tunnetusti laittoi nettiin kuvan, jossa hän poseeraa kylpytakissa parvekkeellaan Kampissa kädessä tukku viiden sadan euron seteleitä. Nyt hän on ruvennut opiskelemaan kauppatieteitä ja on tunnettu sijoittaja,





jonka mielipiteitä arvostetaan finanssialalla.

Niille, jotka vielä saavat nettipokerista elantonsa, pelistä on tullut välineurheilua. Ammattilaisen perustyökalu on ohjelma, joka kerää tilastoja vastustajasta ja näyttää olennaisia lukemia suoraan vastustajan nimimerkin vieressä pelin aikana. On olemassa skriptejä, jotka hälyttävät, kun tunnettu pehmeä pelaaja istuu pelipöytänsä, ja istuttavat automaattisesti ammattipelaajan häntä vastaan.

Lisäksi on olemassa erilaisia offline-työkaluja, joiden avulla voi miettiä puhki menneitä pelitilanteita matematiikan kautta. Uusimpana lisäyksenä ammattilaisen työkalupakkiin on yli tonnin maksava *PioSOLVER*, jolla pystyy laskemaan optimaalisen pelistrategian rajoitetuissa pelitilanteissa. Jos et hanki sitä, voit olla varma, että vastustajasi hankkivat sen, joten kukkaro auki suosiolla. Lisäksi nyt on tullut uutta teknologiaa, joka saattaa tuhota nettipokerin lopullisesti: Vuonna 2017 tekoäly voitti ensimmäistä kertaa huippuammattilaisen No-limit holdemissa.

### Loppu on lähellä

Yksi edellytys nettipokerin toimivuudelle on se, että ihmiset pärjäävät tietokoneille. Jos sivustot ovat täynnä yli-inhimillisiä botteja, ihmisellä ei ole mahdollisuutta voittaa pitkällä aikavälillä, ja rahat voisi käyttää yhtä lailla vaikka rulettiin. Tällainen tilanne

on esimerkiksi backgammon-pelissä, jossa ihmisillä ei ole mitään mahdollisuuksia tietokonetta vastaan. Peliä voi pelata rahasta netissä, ja bottien käyttäminen on kiellettyä, mutta huijaaminen on liian helppoa. Tästä syystä netti-backgammon ei varmasti koskaan tule saavuttamaan samanlaista suosiota kuin nettipokeri.

Nettipokeri on siitä vaikea peli tietokoneille, että siinä on piilotettua informaatiota. Usein voitava peli perustuu siihen, että pääset vastustajan ajatuskulkuihin kiinni ja muokkaat omaa pelityyliäsi sen mukaan. Tietenkin vastustaja yrittää myös parhaansa mukaan päästä sinun pääsi sisään ja mukautua vastaavasti. Peli on psykologista sodankäyntiä, jossa pyritään olemaan aina askeleen edellä vastustajaa. Miten tietokone voisi pärjätä tässä?

Kuuluisa matemaatikko ja taloustieteilijä **John ”kaunis mieli” Nash** todisti, että laajalla joukolla pelejä, joihin myös pokeri kuuluu, on olemassa strategia, jota ei voi voittaa. Tämä strategia ei yritä hyväksikäyttää vastustajan taipumuksia, vaan toimii siten, että mitä tahansa vastustaja yrittää tehdä, tulos on korkeintaan tasapeli vastustajalle pitkällä aikavälillä. Valitettavasti, tai onneksi, tämä strategia on pokerin kohdalla niin monimutkainen ja herkkä, että sitä ei ole mahdollista löytää nykyisellä laskentakapasiteetilla. Silti nykyään monissa pokerivarianteissa tietokone pystyy löytämään strategian,

joka on tarpeeksi lähellä Nashin tasapainostrategiaa, jotta ihmiset eivät enää pärjää sille.

On silti variantteja, joissa tietokone ei vielä pärjää ihmiselle, mutta tekoälyn kehityshistorian valossa on vain ajan kysymys, milloin ihminen jää lopullisesti koneesta jälkeen. Tällä hetkellä eletään luultavasti viimeisiä vuosia, kun maailman paras pokerinpelaaja on ihminen, ja ehkä nettipokeri kokee pian saman kohtalon kuin backgammon. ♣



# KOODAA OMA PELITEKOÄLY

**Häviäminen omalle luomukselleen lempipelissään on täysin omanlaisensa kokemus ylpeyttä ja häpeää.**

Teksti: Lumi Pakkanen

Kuvat: Lumi Pakkanen, Toni Kortelahti

**M**ikäpä olisi luovempaa kuin älyn luominen tyhjästä. Viime aikoina tekoäly on ollut otsikoissa, oli sitten kyse gosta tai Dota 2:sta. Koneoppimiseen perustuvat *AlphaGon* kaltaiset botit ovat vahvoja, mutta kehittäjän rooliksi jää lähinnä luoda riittävät puitteet oppimiseen. Opetusaineiston hankkiminen tai sopivan arkkitehtuurin valitseminen ei aina ole helppoa, mutta yhtä kaikki lopulta syntyy vain ”musta laatikko”, jonka vahvuuksien tai heikkouksien syitä on vaikea ymmärtää.

Siksipä tässä artikkelissa keskitymme ”perinteisiin” lähestymistapoihin luoda pelitekoäly. Laitamme teorian testiin ja implementoimme tekoälyjä *Puyo Puyo* -peliin käyttäen *Monte Carlo* -menetelmiä ja puuhakua.

## Puuhaku

Vuoropohjaiset pelit on mahdollista esittää pelipuuna, jossa jokainen solmu esittää pelitilannetta ja jokainen solmusta lähtevä haara pelitilanteessa tehtyä siirtoa. Koska pelipuun koko on yleensä tähtitieteellisen suuri, sitä ei voida rakentaa kokonaisuudessaan

muistiin. Sen sijaan puu katkaistaan joltain korkeudelta, ja jokaiseen solmuun liitetään arvo, joka kuvaa solmua vastaavan pelitilanteen edullisuutta pelaajalle. Positiiviset luvut kertovat, että ensimmäinen pelaaja on johdolla, ja negatiiviset, että toinen pelaaja on johdolla. Luvun itseisarvo kuvaa etulyöntiaseman suuruutta. Toinen pelaajista pyrkii minimoimaan ja toinen maksimoimaan solmujen arvoa.

Lehtisolmut arvioidaan jonkin yksikertaisen heuristiikan perusteella. Hyvän aseman heuristiikka on oltava yksinkertainen, jotta sen voi laskea nopeasti kaikille saavutetuille lopputilanteille tiettyyn siirtosyvyyteen asti. Esimerkiksi shakissa ”omien nappuloiden arvo miinus vastustajan nappuloiden arvo” on kohtuullinen heuristiikka, kun taas gossa ”omien alueiden arvo miinus vastustajan alueiden arvo” olisi hyvä heuristiikka, mutta se on hyvin vaikea määritellä täsmällisesti. Ongelma on siinä, että gossa oma ”alue” ja sen ”hallinta” eivät ole yksinkertaisesti määriteltäviä käsitteitä.

Kun lehtisolmun arvot on määriteltä, voidaan määritellä arvot puun si-

säsolmuille. Sisäsolmun arvo on joko sen lapsisolmujen maksimi tai minimi riippuen siitä, kumman vuoro on. Tällöin puun juuren arvo kertoo sen hetkisen pelitilanteen arvon. Minmax-puussa käy vain helposti niin, että ratkaisualgoritmin joutuu kirjoittamaan erikseen eri pelaajille. Yleensä kannattaa luoda sen sijaan luoda niin sanottu *negamax-puu*, jossa solmun arvo on maksimi lapsisolmujen arvojen vastaluvuista (negaatioista). Puun solmu maksimoi siis vastustajan häviötä.

Shakissa käytetään yleensä seuraavaa heuristista pisteytystä lautatilanteen arviointiin:

- Sotilas: 1 piste
- Ratsu: 3 pistettä
- Lähetti: 3 pistettä
- Torni: 5 pistettä
- Kuningatar: 9 pistettä

Jos esimerkiksi menettää vaihtokaupassa oman tornin, mutta vangitsee vastustajalta ratsun ja lähetin, jää yhden pisteen voitolle, ja vaihtokaupan voi ajatella olevan edullinen.



## Puyo Puyo

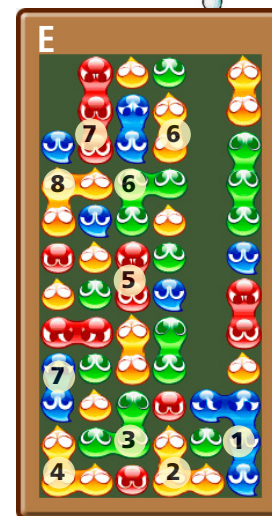
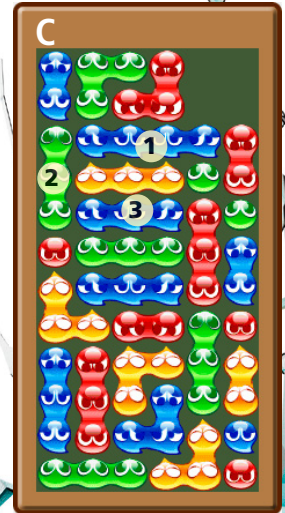
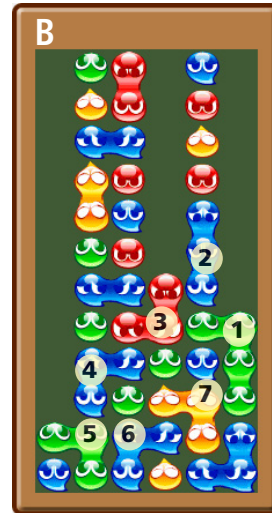
Käytämme tässä artikkelissa esimerkkipelinä Puyo Puyoa, sillä se on botille yhtä aikaa sekä vaikea pelata erinomaisesti että helppo pelata keskinkertaisesti. Pelissä on tarkoitus yhdistää neljä tai useampi samanväristä pelinappulaa yhdeksi ryhmäksi, joka sitten katoaa. Poistetun ryhmän yläpuolella olevat nappulat putoavat ja mahdollisesti yhdistyvät ryhmiksi, jotka katoavat, ja sykli toistuu. Pisteitä saa sitä enemmän, mitä pidemmän ketjun saa rakennettua. Uusia nappuloita tulee laudalle monivärisinä ryhminä, jotka tippuvat kentälle sen ylälaidalta. Peli on siis tavallaan *Tetriksen* ja *Candy Crush / Zookeeper* -tyyppisen pelin sekamuoto.

Yksin pelattavassa Puyo Puyossa ei ole loppua, joten asema on joka tapauksessa arvioitava heuristisesti, koska itse pelipuu on ääretön. Koska pelissä on nähtävissä vain kolme seuraavaa pelinappulaa, on luonnollista tehdä puuhaku kolmen syvyyteen asti. Haku tuntemattomien pelinappuloiden yli onnistuu myös, mutta vaatii huomattavasti enemmän konetehoa, sillä heuristisista arvoista on otettava keskiarvot kaikkien mahdollisten pelinappuloiden yli.

## Monte Carlo

Siinä missä puuhaussa määritellään, mitä tarkoittaa hyvä pelitilanne, Monte Carlossa määritellään, mitä tarkoittaa hyvä siirto. Monte Carlo -puuhaussa pelataan nopeita pelejä tietyllä läpipeluuaktiikalla (policy), joka antaa jokaisessa tilanteessa todennäköisyysjakauman seuraavalle siirrolle. Nyt heuristiikkojen sijaan puun solmujen arvot tulevat läpipeluiden tuloksista. Algoritmissa toistetaan seuraavaa silmukkaa:

1. **Valinta:** Pelipuusta valitaan juuresta lähtien solmuja, kunnes päästään lehtisolmuun asti. Valinta perustuu solmun lupavuuteen tai uutuuuteen. Mitä vähemmän solmua on tutkittu, sitä uudempi se on. Näiden kahden arvon tasapainoa kuvaa algoritmissa *tutkimusvakio* (exploration constant).
2. **Laajennus:** Lehtisolmulle luodaan kaikki lapsisolmut ja niistä valitaan yksi (yleensä satunnaisesti).



- A: Ihmisen rakentama 12-ketju. Numerot kertovat missä järjestyksessä pelinappulat häviävät.  
 B: Tekoälyn rakentama ketju, missä saatavilla olevien ketjujen lukumäärää on maksimoitu.  
 C: Tekoälyn rakentama "ketju", missä on yritetty maksimoida ryhmien kokoja. Ilman signaalia itse pisteityksestä botti saa aikaan lähinnä hyvää muotoa ilman tarkoitusta.  
 D: Tekoälyn rakentama 7-ketju hyödyntäen pelkästään pistesignaalia. Ketju on kohtuullisen pitkä, mutta "sotkuinen" vailla selvää muotoa.  
 E: Tekoälyn rakentama 8-ketju. Evaluaatiossa on maksimoitu kolmen edellisen heuristiikan summaa. Ketjussa on oma outo logiikkansa, mutta tuskin koskaan voi nähdä ihmisen pelaavan tällä tavoin.

3. **Simulaatio:** Pelataan peliä eteenpäin arpoen siirtoja läpipeluuaktiikan jakaumasta.
4. **Takaisinviesti:** Simulaation tulos päivitetään kaikkiin valitun polun solmuihin.

Yllättäen Monte Carlo toimii gohon, sillä vaikka gon pisteityksessä olennainen käsite "hallittu alue" on vaikea, saa pelaaja omalla alueellaan etua, vaikka pelattaisiin täysin satunnaisesti. Valitun läpipeluuaktiikan ei tarvitse siis edustaa vahvaa pelaamista. Tärkeintä on, että peli jatkuu tasaväkisesti ja läpipeluuaktiikka käy siirtoavaruutta läpi tasaisesti.





Toisaalta tietyissä peleissä, kuten shakissa tai Tetrixissä, on hyvin helppo pilata oma asemansa ajattelemattomalla siirrolla, eikä Monte Carlo sovi niihin. Puyo Puyossa taas on vaikea pelata itseään täysin pussiin. Jopa täysin satunnaiseen läpipeluuaktiikkaan perustuva Monte Carlo -botti pelaa hyvää amatööritason peliä.

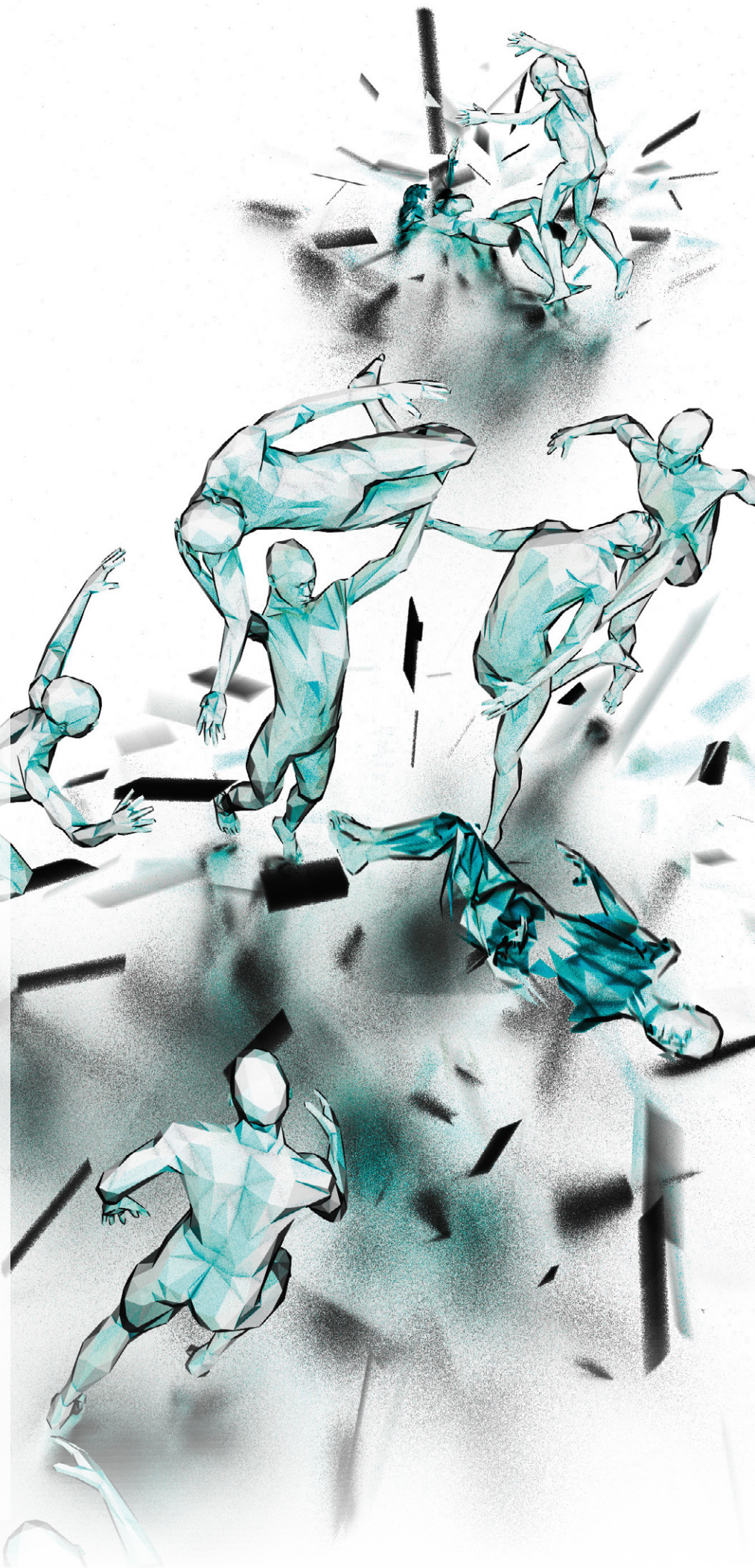
### Moottori kaiken takana

Koska Monte Carlo -puuhaku hyödyntää vain lyhyttä ihmisekspertin määrittelemää läpipeluuaktiikkaa hyvästä pelistä, on ensiarvoisen tärkeää, että taktiikkaa pystytään ajamaan mahdollisimman paljon. Mitä enemmän taktiikkaa voidaan soveltaa pelaajan sisällä, sitä paremmin saadaan esiin siitä syntyvä hyvä keskimääräinen peli. Siispä usein kannattaa mahdollisuuksien puitteissa yksinkertaistaa pelilogiikkaa säilyttäen silti pelin olennaisimmat piirteet, jotta läpipeluu olisi mahdollisimman nopeaa. Samalla myös saadaan pienennettyä haakuvaruuden kokoa.

Pelimoottori määrittelee, millä tarkkuudella peli esitetään, ja kuinka tehokkaasti eri siirtovaihtoehtoja voidaan käydä läpi. Esimerkiksi Puyo Puyossa pelinappulat asetetaan laudalle reaaliajassa, mutta riittävään tarkkuuteen päästään mekaniikalla, jossa nappulat asetetaan ruudun yläreunaan mistä ne tippuvat alas välittömästi. Tämä vähentää eri siirtovaihtoehtoja jopa 70-kertaisesti. Normaalisti nappuloiden tippumiseen menee hieman aikaa, mutta käytännössä tekoäly voi olettaa, että nappulat tippuvat välittömästi, koska ero on pieni. Olennaisinta on luoda pitkiä ketjuja, ja pelimoottori kannattaa suunnitella tukemaan vain sitä.

Kun kyseessä on diskreetissä ruudukossa pelattavasta pelistä, on usein tehokkainta esittää pelitila bittiruudukoilla. *Bittilauta* on kokonaisluku, jonka bitit vastaavat ”kyllä tai ei” kysymyksiin jokaisen pelilaudan ruudun kohdalta. Puyo Puyossa olennaiset kysymykset ovat, että onko tietyssä ruudussa tietyn värinen nappula.

Bittilautaesityksen hyöty on siinä, että kaikkia laudan nappuloita voidaan käsitellä kerralla yksinkertaisilla bittioperaatioilla. Esimerkiksi, jos laudan leveys on 8 pelinappulaa, saadaan sopivasti määritellyllä bitti-





laudalla siirrettyä kaikkia tietyn värin nappuloita yksi ruutu alemmas operaatiolla  $f(x) := x \ll 8$ .

Oikealla esimerkki yksinkertaistusta Puyo Puyon painovoima-algoritmista, joka tukee vain yhtä väriä.

## Pelaamaan

Kokeilimme neljää eri bottia Puyo Puyon yksinpelissä. Yksinkertaisin näistä on satunnaista peluutaktiikkaa käyttävä yhden simulaation Monte Carlo -botti. Tämä botti arvioi pelipuun lehden arvon yhden satunnaisen jatkopelin perusteella. Annoimme tälle kuitenkin sen verran apua, että se ei tee itsemurhasiirtoja. Tämä botti onnistui lopulta saamaan noin 14,4 pistettä siirtoa kohden. Ilman suunnittelua ketjujen pituus jäi yleensä alle kolmen. Toki jopa kymmenen pituisia ketjuja saattaa syntyä vahingossa, mutta sen todennäköisyys on luokkaa yksi sadastatuhannesta.

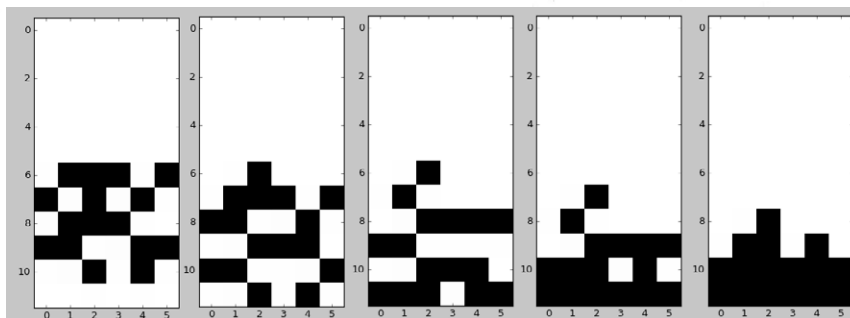
Kun satunnaista läpipeluuaktiikkaa ajetaan 10 000 kertaa per siirto, pelin laatu paranee huomattavasti. Kymmenellä tuhannella simulaatiolla siirtoa kohden päästiin tulokseen 351,52 pistettä siirtoa kohden. Ilman mitään ymmärrystä muodosta Monte Carlo jää kuitenkin amatöörin tasolle.

Hyvin pärjasi kuitenkin myös puhtaaseen pistesignaaliin eli pelkästään tiettyssä siirrossa välittömästi ansaittuihin pisteisiin perustuva viiden syvyinen puuhaku. Tämä menetelmä sai aikaan keskimäärin 649,5 pistettä per siirto. Sokea pisteiden kerääminen harhautti botin keräämään paljon yhden pitkiä ketjuja, mutta keskimääräinen ketjun pituus oli kuuden paikkeilla.

Parhaiten testissämme toimi neljän syvyinen puuhaku kolmen heuristiikan summalla, keräten jopa 943,3 pistettä siirtoa kohden. Käsin ohjelmoituiden piirteet ohjaavat bottia hyvään muotoon, vaikka kokonaiskuva puuttuikin. Ketjut ovat johdonmukaisesti yli kymmenen mittaisia.

## Gladiaattorit areenalle

Peluutimme myös kaikkia neljää bottia toisiaan vastaan kaksinpelissä. Moninpelissä ketjut luovat pisteiden sijaan haittanappuloita vastustajan laudalle. Pelin häviää, jos ei pysty tekemään enää siirtoa. Bottien voittotodennäköisyydet toisiaan vastaan on listattu taulukossa 1.



Bitteihin perustuvassa painovoimassa nappula pudotetaan vain, jos sen alla on tyhjää. Tästä seuraa suurempien rakenteiden "pisaroituminen", missä koko pino hajoaa ennen kuin se asettuu ruudukon pohjalle.

```
WIDTH = 6
HEIGHT = 12
BOTTOM = sum(
    1 << (i + WIDTH * (HEIGHT - 1))
    for i in range(WIDTH)
)

def handle_gravity(puyos):
    temp = 0
    while temp != puyos:
        temp = puyos
        below = (puyos >> WIDTH) | BOTTOM
        falling = puyos & ~below
        puyos = (falling << WIDTH) | (puyos & below)
    return puyos
```

Listaus: painovoiman toteuttaminen bittiopeeraatioilla.

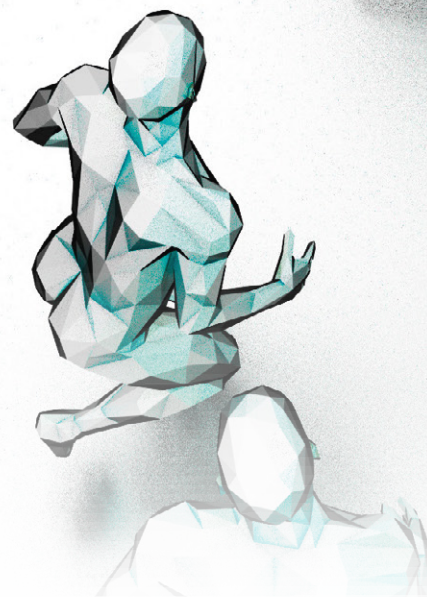
	satunnainen	pistesignaali	Monte Carlo	heuristiikka
Satunnainen Monte Carlo yhdellä simulaatiolla	50 %	0 %	0 %	0 %
pistesignaali	100 %	50 %	98 %	40,5 %
Satunnainen Monte Carlo 10 000 simulaatiolla	100 %	2 %	50 %	6,25 %
heuristiikka	100 %	59,5 %	93,75 %	50 %

Taulukko 1.

Satunnainen Monte Carlo yhdellä simulaatiolla häviää kaikille muille boteille, mikä on lievästi yllättävää, sillä allekirjoittanut on silloin tällöin kompastellut sitä vastaan. Kymmenen tuhannen simulaation Monte Carlon ongelma on aivoton läpipeluuaktiikka – simppeillä peruspelillä ei voita. Naivistisesti seuraavan hyökkäyksen kokoa optimoiva pistebotti pärjää yllättävän hyvin, ja allekirjoittaneen paras heuristiikkaresepti nousee vain 10 prosenttiyksikköä sen yläpuolelle.

## Loppusanat

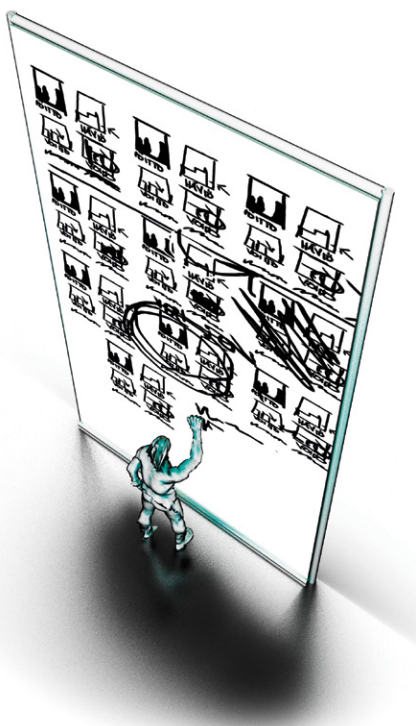
Oman tekoälyn ohjelmointi on hauskaa puuhaa, varsinkin jos pitää myös pelimoottoreiden kehittämisestä. Tässä artikkelissa



esitettyjen algoritmien pelivahvuus riippuu suoraan siitä, paljonko eri siirtosekvenssejä ehditään käydä läpi. Allekirjoittaneella oli vaikeuksia keksiä hienostuneita heuristiikkoja, joten ymmärryksen syvällisyys oli korvattava laskemalla syvemmälle pelipuuhan. Valitettavan usein raaka laskentavoima jyrää nokkelat ja näppärät käsin kudo-  
tut taktiikat. ♁

### Bottinikkarin muistilista

- Yksinkertaista pelin sääntöjä, jotta hakuvaruus pienenty.
- Sano se bitein. Bittilaudat ovat usein tehokas tapa toteuttaa pelimoottori.
- Lyö sitä Monte Carlolla. Jo täysin satunnaisen pelin voimasuhteet kertovat paljon.
- Etsi puutteet. Tekeekö MC-botti jotain systemaattisesti väärin?
- Paranna Monte Carlon läpipeluu-  
tietokantaa. Suodata pois selvästi väärät siirrot.
- Onko MC-botilla edelleen puutteita? Ehkä puuhaku on parempi ratkaisu.
- Kehitä puuhakuun sopivia heuristiikkoja. Pohdi mikä tekee asemasta vahvan.
- Bonus: Tee pelin minimaaliseen versioon täydellinen pelaaja. Voiko siitä oppia jotain? Voiko taulukkoa käyttää heuristiikkojen arvioimiseen?



### Jumala taulukossa

Yksinkertaisimmat pelit voidaan ratkoa tehokkaasti käymällä läpi kaikki mahdolliset pelitilanteet. Kun esimerkiksi shakissa laudalla on enää muutama nappula, on mahdollista ratkaista peli täydellisesti keräämällä kaikki pelitilanteet niin sanottuun *loppupelitaluk-  
koon*.

Loppupelitalukko on vanha idea peräisin vuodelta 1965. Sen sijaan, että parasta siirtoa etsittäisiin pelaamalla peliä eteenpäin, tehdäänkin haku pelin lopusta alkaen. Ensinnäkin listataan kaikki voitettavat pelitilanteet, sitten kaikki pelitilanteet, jotka johtavat suoraan voittoon, sen jälkeen kaikki pelitilanteet, jotka johtavat näihin, ja niin edelleen. Lopulta on saatu listattua kaikki voitettavat pelitilanteet. Pelitilanteet, jotka eivät ole listassa, ovat tasapelejä.

Tämän lähestymistavan ongelma on lähinnä se, että eri pelitilanteita on valtavan paljon. Niinpä se sopii lähinnä muutaman nappulan loppupelien shakissa tai hyvin pienten lautojen ratkaisemiseen gossa.

Yllä kuvattuun retrogradiseen analyysiin perustuva algoritmi on verrattain yksinkertainen. Kehittäjän kekseliäisyyttä tarvitaan lähinnä datan pakkaukseen ja tietorakenteen kehittämiseen, jolla löydetty pelitulos saadaan liitettyä oikeaan pelitilanteeseen mahdollisimman pienellä määrällä muistia.

Ongelma pelkässä voiton listaamisessa on se, että vaikka tiedettäisiinkin, että tietty pelitilanne johtaa voittoon ja nämä siirrot ylläpitävät voittoasemaa, ei tiedetä, mikä siirto edistää voiton saavuttamista. Siksi pelkän voiton lisäksi listataan usein myös etäisyys voittoon. Paras siirto on se, jonka etäisyys voitosta on yhden pienempi kuin nykytilanteen.

Tässäkin lähestymistavassa on ongelmansa, jos molempien pelaajien voittomahdollisuuksia listataan samassa taulukossa. Kun voitava pelaaja yrittää mattia mahdollisimman nopeasti, yrittää altavastaaja jarruttaa mattia mahdollisimman tehokkaasti, mikä tekee ongelmasta epäsymmetrisen. Siksi jokainen pelitilanne sisältää itse asiassa kaksi eri tulosta ja kaksi eri etäisyyttä:

Kuinka paljon pisteitä voidaan saavuttaa ja kuinka vähäisessä määrässä siirtoja.

Kuinka pitkään vastustajan pisteiden saamista voidaan jarruttaa ja kuinka vähän pisteitä hän silloin saa. Shakissa pisteitä ovat vain +1 (voitto), 0 (tasapeli) ja -1 (häviö).

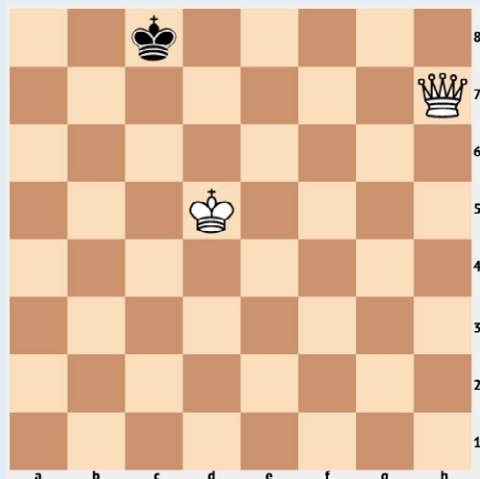
Yllättäen jälkimmäinen tavoite on tietyssä mielessä edullisempi. Shakissa on jopa mahdollista estää vastustajaa saamasta pisteitä lainkaan, mikä lasketaan siis äärettömän hyväksi jarruttamiseksi.

Mainittakoon, että gossa tämä ero on erityisen tärkeä, sillä yksinkertaista voittoehto ei ole, ja pelissä syntyy kehämisiä tilanteita, joissa kumpikaan pelaaja ei itsestään selvästi yritä "matittaa". Kehien rikkomiseksi on välttämätöntä tarkastella viivytysstrategiaa ja sen estämistä vastastrategialla.

Loppupelitalukon avulla pelin voittoehto voidaan määritellä botille uusiksi. Esimerkiksi shakissa sen sijaan, että peli päättyy kuninkaan matittamiseen, voidaan määritellä, että peli päättyy, kun päädytään tilanteeseen, joka löytyy valmiiksi laskettuna loppupelitalukosta. Näin saadaan korvattua kokonaisia pelipuuhaa valmiiksi lasketulla lopputuloksella. Shakkia varten valmiiksi laskettuja kuuden nappulan loppupelitalukoita löytää netistä helposti ilmaiseksi.

Tällaisen voi helposti liittää omaan bottiin ja nostaa sen pelin seuraavalle tasolle.

Valkea voi tässä tilanteessa pakottaa matin kolmessa puolisiirrossa (Kc6 Kd8 Qd7) tai pakottaa pelin jatkumaan loputtomiin. Mustalla ei ole keinoja vangita valkean kuningatarta, ja valkean tulos on viivytysmielessä "äärettömän hyvä".







# PELI FARMI

*Make your winning move.*

## Pelialan valmennusohjelma

17–29-vuotiaille / kesto 3 kk

Haluatko oppia tekemään pelejä tai oletko jo hankkinut osaamista grafiikan, koodaamisen tai vaikka käsikirjoittamisen ja pelisuunnittelun saralta? Mietitkö, löytyisikö pelialalta sinulle sopiva työ- tai opiskelupaikka?

Hae mukaan kevään valmennusohjelmaan sivujemme kautta – haku auki nyt!

**READY TO PLAY? [www.pelifarmi.fi](http://www.pelifarmi.fi)**

### PLAYER 1

17–29-vuotias

### SITE

Oulu & Tampere

### TEAM:



6Aika



Vipuvoimaa EU:lta 2014–2020

OULU



TAMPERE

# Funktionaalista ohjelmointia Clojurella

*Funktionaalinen ohjelmointi tarjoaa elegantit työkalut suurten koodimassojen hallintaan.*

Teksti: Robert Brotherus

Kuvat: Sakari Leppä, Robert Brotherus, Nasu Viljanmaa

**O**hjelmoinnin haasteena on pitää hallinnassa monimutkaisuus. Yksinkertaisen ohjelman kasvaessa pikkuhiljaa satoihintuhansiin riveihin uusien ominaisuuksien lisäämisestä vanhoja rikkomatta tulee yhä vaikeampaa.

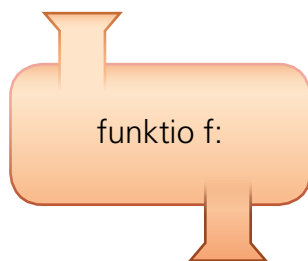
Tyypillinen bugi on, että ohjelman globaalit muuttujat tai jonkin olion kentät ovat menneet outoon tilaan ja ohjelmoijan pitää selvittää, miten tähän on päädytty. Debuggaus edellyttää salapoliisityötä oudosta tilasta askel askeleelta taaksepäin, jotta saadaan selville mitä, milloin ja missä meni alun perin pieleen. Kun ongelman pohjimmainen syy on lopulta löydetty ja tehdään korjaus, on yleistä, että korjaus hajottaa jonkin toisen ohjelman toiminnon. Näiden ongelmien selvittäminen on vaivalloista, ja ne tekevät laajan ohjelman kehittämisestä ja ylläpidosta kallista.

## Ei muuttujia, ei muuttujaongelmia

Funktionaalinen ohjelmointi on yleisesti käytetylle olio-ohjelmoinnille sekä perinteiselle imperatiiviselle ohjelmoinnille vaihtoehtoinen tyyli, joka pyrkii helpottamaan edellä kuvattuja

ongelmia. Funktionaalinen ohjelma pyritään koostamaan funktioista, joilla ei ole *sivuvaikutuksia*. Tällaiset niin sanotut *puhtaat funktiot* ottavat parametreja ja palauttavat niistä lasketun tuloksen, mutta eivät muuta ohjelman *tilaa* eli erilaisten muuttujien arvoja millään tavalla. Puhtaiden funktioiden ymmärtäminen, testaaminen ja debuggaaminen on selkeää ja niiden uudelleenkäyttö helppoa. Kaikenlaisien muuttujien käyttö pyritään minimoimaan.

syöte  $x$



tuote  $f(x)$

Puhdas funktio. Funktionaalinen ohjelma pyritään koostamaan lähes yksinomaan puhtaista funktioista, jotka ottavat parametreja ja palauttavat lopputuloksen, joka riippuu vain parametreista. Puhdas funktio ei saa lukea tai muuttaa ohjelman tilaa, tietokantaa, levytiedostoja tai muuta sellaista.

Rinnakkaistaminen on tämän päivän kuumimpia ohjelmointitrendejä, koska prosessoreiden kelloaajuudet ovat jo usean vuoden ajan polkeneet paikallaan, ja siten ainoa merkittävä keino nopeuttaa ohjelmaa on lisätä ytimien määrää ja rinnakkaisuutta. Muuttujien muuttamiseen perustuvan ohjelmoinnin ongelmat moninkertaistuvat, kun koodiin päästetään mellaamaan yhtä aikaa monta säiettä, jotka muuttuvat ja lukevat muuttujia. Näiden ongelmien välttäminen puolestaan vaatii monimutkaisia ja itsessään bugiherkkiä lukitusratkaisuja.

Onnistuessaankin lukitukset joutuvat usein siihen, että säikeet odottelevat valtaosan aikaa lukkojen vapautumista, ja näin menetetään suuri osa rinnakkaisuuden potentiaalisesta tehohyödyistä. Puhtaita funktioita voi sen sijaan aina kutsua vapaasti rinnakkain missä järjestyksessä vain, ilman että eri funktioiden suoritukset sotkevat toisiaan.

## Tilan hallinta Clojuressa

Puhdas funktio ottaa sisään dataa ja tuottaa siitä (ja vain siitä) tuloksen. Puhtaan funktionaalinen ohjelma koostetaan toisiaan kutsuvista puhtaista funktioista, joten kokonaisuutena tällainen ohjelma on yksinkertaisesti yksi iso funktio, joka ottaa sisään da-



taa, tuottaa siitä tuloksen ja loppuu siihen.

Tarkka lukija ymmärtää, että kovin monet oikean maailman ongelmia ratkovat ohjelmat eivät voi olla puhtaan funktionaalisia. Näihin harvoin, jotka voivat olla puhtaan funktionaalisia, lukeutuvat esimerkiksi kääntäjät: ne syövät joukon lähdekoodia ja tuottavat suoritettavan binääritiedoston. Suurin osa ohjelmista kuitenkin tarvitsee jonkinlaista ajanhetkestä toiseen muuttuvaa tilaa ollakseen hyödyllisiä. Clojuren vastaus tähän haasteeseen seuraava:

1. Kuvataan koko ohjelman tila vain yhdellä globaalilla muuttujalla, jonka arvo koostuu sisäkkäisistä Clojuren perustietotyypeistä. Tämä tila ei sisällä mitään dataa, jota tarvitaan vain väliaikaisesti.
2. Kun ohjelman tilan pitää muuttua, niin tila-muuttujan arvo vaihdetaan aina kerralla kokonaan uuteen arvoonsa. Jos esimerkiksi tila sisältää kahden pankkitilin A ja B saldon, niin tilisiirron jälkeinen uusi tila sisältää A:n kasvaneen arvon ja B:n pienentyneen arvon. Tila-muuttujan arvo ei koskaan sisällä ristiriitaisia välimuotoja eri tilojen välillä. Uuden tila-arvon valmistamisessa voi olla useita vaiheita, mutta sen hoitaa joukko puhtaita funktioita, jotka eivät koske tila-muuttujaan.
3. Tila-muuttuja pidetään Clojuren *atom*-tallennuspaikassa, joka mahdollistaa tilan turvallisen vaihtamisen uuteen monisäikeisessä ohjelmassa. Kaikki funktiot näkevät aina vain yhden version tilasta. Tila vaihdetaan uuteen antamalla funktio, joka ottaa vanhan tilan parametrina ja palauttaa uuden tilan. Koko ohjelma sisältää vain yhden tällaisen tilaa vaihtavan sivuvaikutuksellisen funktion. Kaikki muut funktiot ovat puhtaita, joten suuressa ohjelmassa lähes 100% funktioista on puhtaita.

Tämä ohjelmointimalli ei ota kantaa serialisointiin eli tilan pysyvämpään tallennukseen levyille esimerkiksi tietokantaan. Omissa Clojure-harrasteprojekteissani olen yleensä ohjelman tilan muuttuessa samalla tallentanut tilan serialisoituna Clojure-tekstinä tiedostoon. Näin ohjelmaa voidaan jatkaa seuraavalla käynnistyskerralla samasta

## Clojuren kokoelmatyyppejä

**Vektori:** [ 1, 2, [ 2.5, 3.4 ], 3 ]

**Lista:** ( 1, 2, 3, "matti" )

**Hakemisto:** { :matti 32, :mikko 20, :seppo 17 }

tilasta. Tämä ei vaadi juuri ollenkaan ylimääräistä aikaa tai vaivaa, koska Clojure-tietorakenteiden serialisointi, deserialisointi ja tallennus voidaan kaikki tehdä yhdellä rivillä ja rinnakkaisessa säikeessä.

Tila voidaan toki tallentaa myös tietokantoihin, joista parhaiten Clojuren pariin sopii Clojuren tekijän **Rich Hickeyn** kehittämä **Datomic**-tietokanta. Datomic-kannassa ei koskaan *muuteta* olemassa olevaa tietoa, vaan luodaan vain uusia tietoja. Datomic laajentaa näin muuttumattomien tietotyyppien idean tietokantojen maailmaan.

Edellä kuvatun tyyppisessä ohjelmassa myös monisäikeisyys on hyvin luontevaa. Kaikki tilaa lukevat ja siitä erilaisia tuloksia tuottavat säikeet (esimerkiksi http-kyselyyn vastaava html-näkymän muodostava funktio) voivat rauhassa käyttää tilan eri attribuutteja ilman pelkoa, että ne olisivat ristiriidassa keskenään, tai että ne muuttuisivat funktion suorituksen aikana. Clojure pitää tilaa päivittävät funktiot automaattisesti järjestyksessä niin, että vain yksi kerrallaan päivittää tilaa.

Myös undo-toiminto saadaan käytännössä ilmaiseksi tällaisessa arkkitehtuurissa. Aina kun ohjelman tila vaihdetaan uuteen, voidaan edellinen tila tallentaa historialistaan ja hakea se sieltä myöhemmin, kun halutaan palata aiempaan tilaan.

## Clojuren tietotyypit – miltei liian hyvää ollakseen totta

Koska puhtaassa funktionaalisessa ohjelmoinnissa ei käytetä muuttujia, kaikki tietorakenteet ovat muuttumattomia (*immutable*). Tilanteissa, joissa olio-ohjelmoinnissa muutettaisiin eli mutatoitaisiin objektin kenttiä, luodaan funktionaalisessa ohjelmoinnissa uusi tietorakenne, joka sisältää halutut muutokset. Tämä saattaa kuulostaa hankalalta, hitaalta ja paljon muistia vievältä ratkaisulta, ja näin saattaa toki ollakin. Hyvä funktionaalinen ohjelmointikieli tarjoaa kuitenkin tietorakenteet ja menetelmät, joilla uuden version tekeminen objektista on helppoa ja nopeaa sekä vie vain vähän muistia.

Funktionaalisten kielten esiäidissä LISPissä kaikki tietorakenteet perustuivat linkitettyihin listoihin, joista on nopeaa (aikavaativuus  $O(1)$ ) tehdä uusi pidempi versio ilman, että alkuperäinen lista muuttuu tai että listan alkiot kopioidaan muistissa. Linkitetty lista on kuitenkin paljon hitaampi ( $O(N)$ ), kun pitää esimerkiksi hakea alkio indeksillä tai tutkia, sisältääkö lista jonkin tietyn alkion. Osin tästä johtuen perinteisiä LISP-kieliä on syystä pidetty melko hitaina oikean maailman ongelmiin.

Moderni Clojure sen sijaan sisältää joukon kokoelmatyyppejä kuten listat, vektorit ja hakemistot (*map*). Näistä löytyy uuden objektin luova nopea ( $O(\log_{32}N)$ ) eli käytännössä lähes vakioaikainen ja vähän muistia kuluttava operaatio kaikkiin yleisiin tarpeisiin. Tämä saavutus voi kuulostaa liian hyvältä ollakseen totta mutta on mahdollinen kekseliään implementaation avulla. Clojure käyttää niin sanottuja persistenttejä tietorakenteita, joissa uuden tietorakenteen toteutus viittaa alkuperäisen rakenteen tietoihin niiltä osin kuin ne ovat samoja. Tämä on turvallista, koska alkuperäinen data ei voi koskaan muuttua. Tehokas funktionaalinen ohjelmointi on usein mahdollista ilman näitä tehokkaita tietotyyppiejä.

## Funktionaalinen työkalupakki

Kaikenlaiset muuttujat globaaleista paikallisiin tuntuvat joskus hyvin käteviltä moneen tarkoitukseen, ja muuttujien välttäminen tekee ohjelmoinnista omalla tavalla haasteellisempää. Funktionaalisessa tyyliässä vaaditaankin joskus hieman enemmän aivotyötä jonkin ohjelmointiongelman ratkaisemiseen. Lopputulos on kuitenkin yleiskäyttöisempi, kestävämpi ja vähemmän virhealtis kuin muuttujiin perustuva ratkaisu. Mitä pitkäkestoisempi ja laajempi ohjelmointiprojekti siis on kyseessä, sitä enemmän investointi funktionaaliseen tyyliin maksaa itseään takaisin, joskus jopa moninkertaisesti.

Funktionaalisessa tyyliässä muuttujien sijaan käytetään toisenlaisia



Kiinan kielen kirjoitusta Clojurella.

työkälyä kuten *rekursio*, *funktio-objektit* ja *korkeamman asteen funktiot*. Rekursiolla, itse itseään kutsuvalla funktiolla, voidaan toteuttaa toimintoja, jotka toteutettaisiin muissa tyyliissä silmukkarakenteilla, joiden tila muuttuu iteraatiosta toiseen.

Funktionaalissa tyyliissä funktioita voidaan käsitellä kuten mitä tahansa objektia. Funktiot voivat ottaa parametreina toisia funktioita tai palauttaa tuloksena funktioita. Monet yleiskielletkin ovat alkaneet viime vuosina tulla näitä menetelmiä, mutta kun muuttujia ei käytetä lainkaan, nämä menetelmät nousevat tärkeämpään rooliin.

Funktionaalista ohjelmointia helpottaa myös se, että kaikki data voidaan pitää avoimesti kaiken ohjelmakoodin näkyvissä sen sijaan, että sitä piiloteltaisiin privaattikenttiin, kuten olio-ohjelmoinnissa suositellaan. Kun dataa voi vain lukea eikä muuttaa, näkyvyyksiä ei tarvitse rajoittaa.

## Dynaaminen tyyppitys

Dynaamisesti tyyppitetyissä kielissä, kuten Clojuressa, tyyppiä ei tarvitse määrittellä etukäteen. Tämä vähentää komponenttien välistä riippuvuutta tehden funktioista yleiskäyttöisempiä ja mahdollistaen aina helpon serialisoinnin. Ilman erillisiä tyyppimäärittelyjä tarvitaan myös vähemmän koodirivejä ja päästään nopeammin liikkeelle ohjelman kirjoittamisessa. Tarvittavat tietorakenteet kehittyvät pikkuhiljaa käsi kädessä niitä käsittele-

vien funktioiden kanssa.

Clojurea pidetäänkin yhtenä kaikkein ilmaisuvoimaisimmista kielistä, eli pienellä määrällä koodirivejä voi saada paljon aikaan. Tämä etu korostuu laajoissa ohjelmissa koodin paremman uudelleenikäytävyyden ansiosta. Toki usein luettavuuden vuoksi voi olla syytä kirjoittaa koodi hieman laajemmin kuin lyhin mahdollinen ilmaisu.

Dynaamisen tyyppityksen varjopuoleksi voidaan lukea se, että kääntäjän on vaikeampi optimoida koodia, jolloin suorituskyky on usein staattisesti tyyppitettyä kieltä heikompi. Clojure on kuitenkin dynaamiseksi kieleksi merkittävän nopea. Se voittaa useilla alueilla esimerkiksi Rubyn, Pythonin ja JavaScriptin. Lisäksi yleensä moderneissa suurissa ohjelmissa vain pieni osa ohjelmasta on suorituskyvyn kannalta kriittistä. Koska JVM-pohjaisesta Clojuresta on helppo kutsua Java-funktioita, voidaan tällaiset toiminnot tarvittaessa koodata Javalla (tai C:llä Java-C-linkin kautta). Tällöin toki ohjelman funktioiden puhtaus on vaarassa kärsiä.

Dynaamisesti tyyppitetyissä kielissä osa tyypeihin liittyvistä bugeista tulee esille vasta ajon aikana eikä käännösvaiheessa. Tähän haasteeseen merkittävin ratkaisu ovat automaattiset (yksikkö)testit, jotka toki ovat tärkeitä, vaikka kieli olisikin staattisesti tyyppitetty. Yksikkötestien tekeminen edellä kuvatussa Clojure-arkkitehtuurissa onkin hyvin suoraviivaista, kos-

ka puhtaan funktion testaaminen ei vaadi minkäänlaisen tilan rakentelua, ja ohjelman funktioista 99 prosenttia on puhtaita. Clojure sisältää myös kirjastoja, kuten *spec*, jotka mahdollistavat funktioiden parametrien tarkemman tyyppin määrittelyn uhrattamatta dynaamisen tyyppityksen etuja.

Syntaksin osalta Clojure, kuten muut LISPit, poikkeavat radikaalisti muista kielistä siten, että kielen syntaksi koostuu kielen omista tietorakenteista. Clojure-ohjelma siis rakentuu listoista, vektoreista ja hakemistoista, joilla kuvataan ohjelman funktiot. Tämä ja Clojuren niin sanotut *makrot* mahdollistavat vertaansa vailla olevan *metaohjelmoinnin* eli ohjelman, joka tarpeen mukaan generoi itse itseään. Tämä vähentää entisestään koodin tarpeetonta toistoa, mikä helpottaa koodin ylläpitoa. Makroilla voidaan myös laajentaa kieltä dynaamisesti uusilla ominaisuuksilla, jotka muissa kielissä pitää kovakoodata kielen syntaksiin ja kääntäjään.

## Clojurea ja kiinaa

Seuraavassa esittelen omista Clojure-harrasteprojekteistani joitakin funktioita, jotka valaisevat edellä esiteltyjä Clojuren piirteitä. Tämän artikkelin puitteissa ei ole mahdollista käydä läpi monia Clojuren perusteita, joten koodiesimerkit saattavat avautua paremmin, jos tutustut perussyntakseihin esimerkiksi sivustolta [clojure-doc.org/articles/tutorials/introduction.html](https://clojure-doc.org/articles/tutorials/introduction.html).

Omissa projekteissani olen pyrkinyt yhdistämään hyötyä ja huvia. Kiinan kieltä opiskellessani tein kiinan kielen kirjoitusohjelman <https://github.com/rbrother/ChineseWriter>, jolla voi kirjoittaa ja lukea kiinaa käyttäen äännekirjoitusta ja sisäänrakennettua 100 000 sanan sanakirjaa. Ohjelman käyttöliittymä on tehty Windowsin WPF-kirjastolla ja C#-kielellä, mutta sovelluslogiikka ja tilan hallinta ovat kokonaan Clojuren puolella, tarkemmin sanottuna sen C#-yhteensopivalla clojure-clr-versiolla. Ohjelma on monisäikeinen, jolloin sanakirjasta voidaan jatkuvasti hakea kirjoitettuun tekstiin sopivia kirjainmerkkejä ilman, että käyttöliittymä hyytyy.

**Def**-lauseella annetaan Clojuressa symbolille arvo. Seuraavassa luodaan **state**-symboli, jonka arvoksi tulee kirjoitettavan tekstin tilan ylläpitävä

```
(def state (atom { :text [] :cursor-pos 0 })))
```

Listaus 1.

```
(defn insert-words! [ words ] (swap! state insert-words words) )
```

Listaus 2.



```
(defn insert-words [ { :keys [ text cursor-pos ] } new-words ]
  (let [ new-text (concat (take cursor-pos text) new-words (drop cursor-pos text))
        new-cursor-pos (+ cursor-pos (count new-words)) ]
    { :text new-text :cursor-pos new-cursor-pos } ))
```

### Listaus 3.

```
{ :english "next week" :hanyu "下个星期" :hanzi-rarity 1698 :known 4 :pinyin "xia4 ge4 xing1 qi1"
:short-english "next week" }
{ :english "delicious, tasty" :hanyu "美味可口" :hanzi-rarity 2762 :known 1 :pinyin "mei3 wei4 ke3
kou3" :short-english "delicious" }
{ :english "mineral spring water" :hanyu "矿泉水" :hanzi-rarity 12291 :hsk-index 1178 :known 1 :pinyin
"kuang4 quan2 shui3" :short-english "mineral spring water" }
{ :english "No smoking, Please do not smoke" :hanyu "请勿吸烟" :hanzi-rarity 28690
```

### Listaus 4.

```
(defn load-database [ cc-dict-file short-dict-file ]
  (let [ short-dict (create-hanyu-pinyin-dict (load-from-file short-dict-file))
        large-dict (create-hanyu-pinyin-dict (load-from-file cc-dict-file))
        full-dict (merge-with merge large-dict short-dict)
        merged-words (sort-suggestions (vals full-dict))
        hanyu-indexed (index merged-words [ :hanyu ])
        sort-and-simplify (fn [word-list] (map simple-props (sort-suggestions word-list))) ]
    (reset! info-file-name short-dict-file)
    (reset! all-words (vec (map simple-props merged-words)))
    (reset! hanyu-pinyin-dict full-dict)
    (reset! hanyu-dict (map-map-keys-values :hanyu sort-and-simplify hanyu-indexed ))))
```

### Listaus 5.

```
(defn find-words [ input english ]
  (let [ matcher ((if english english-matcher pinyin-matcher) input) ]
    (take 500 (filter matcher @all-words))))

(defn pinyin-matcher [ pinyin-start ]
  (fn [ { pinyin :pinyin } ]
    (let [ pinyin-no-tones (-> pinyin (str/replace #"[ : ]" ""))
          str/lower-case remove-tone-numbers ] )
      (starts-with pinyin-no-tones pinyin-start))))
```

### Listaus 6.

**atom**-muuttuja. Muuttujan alkuarvona on hakemisto, joka sisältää tyhjän tekstin ja kursorin paikan 0. [Ks. listaus 1.]

Seuraavassa määritellään **defn**-lauseella **insert-words!**-funktio, jolla lisätään kirjoitettavaan tekstiin uusia sanoja. Nimessä käytetään huuto-merkkiä kertomaan, että se kuuluu ohjelman harvoihin sivuvaikutuksellisiin funktioihin. Funktio ottaa parametrina lisättävät sanat (words) ja kutsuu Clojuren **swap!**-funktioita, jolla vaihdetaan atom-muuttujan tila uuteen hallitusti. **Swap!** ottaa parametreina (1) atom-muuttujan (**state**), (2) funktion, jolla vanhasta tilasta tehdään uusi tila (**insert-words**) ja (3) parametrit tuolle funktiolle (**words**). [Ks. listaus 2.]

Edellisen käyttämä **insert-words** on puhdas funktio, joka ottaa parametreinaan ohjelman tilan ja uudet sanat. Ensimmäisestä parametrasta puretaan **text**- ja **cursor-pos**-arvot. Funktio käyttää **let**-rakennetta uuden tekstin ja uuden kursorin sijainnin laskemiseen ja sen jälkeen muodostaa tulokseksi

uuden hakemiston näistä uusista attribuuteista. [Ks. listaus 3.]

Listauksessa 4 on osa ohjelman sanakirjaa, joka on vektori sanoja kuvaavia hakemistoja.

Sanakirja jakautuu 100 000 sanan staattiseen sanakirjaan ja pieneen dynaamiseen sanakirjaan, joka pitää sisällään käyttäjän lisäämät ja muokkaamat sanat. Seuraavassa **load-database**-funktiossa nämä ladataan tiedostoistaan (**short-dict** ja **large-dict**) sekä yhdistetään **merge-with merge**-operaatiolla, joka yhdistää sekä eri sanakirjojen sanat että sanojen attribuutit. Nopeaa hakua varten yhdistetystä sanakirjasta tehdään eri avainten mukaan indeksoituja hakemistoja **index**-funktioilla sekä sanojen yleisyyden mukaan järjestetty versio **sort-suggestions**-funktioilla. Lopuksi **reset!**-funktioilla nämä listat ja hakemistot tallennetaan atom-muistipaikkoihin käyttöä varten. [Ks. listaus 5.]

Sanoja haetaan **find-words**-funktiossa yhdistetystä yleisyyden mukaan

```
(defn new-game-state [ ]
  { :counter 0
    :map {}
    :planets {}
    :units {}
    :players {} } )
```

### Listaus 7.

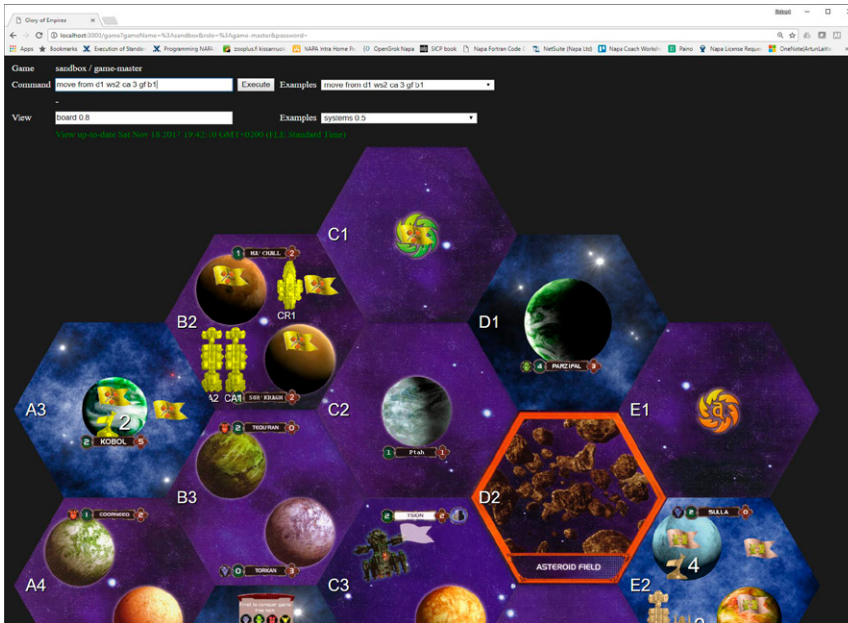
järjestetystä sanakirjasta englannin tai kiinan (pinyin) hakusanan mukaan. Haku tapahtuu taustasäikeessä, ja koska Clojuren **filter**-funktio on ”laiska”, sen ei tarvitse kahlata läpi koko 100 000 sanan listaa, vaan se hakee lisää osumia sitä mukaa, kun käyttöliittymä niitä pyytää. [Ks. listaus 6.]

## Avaruussotaa verkossa

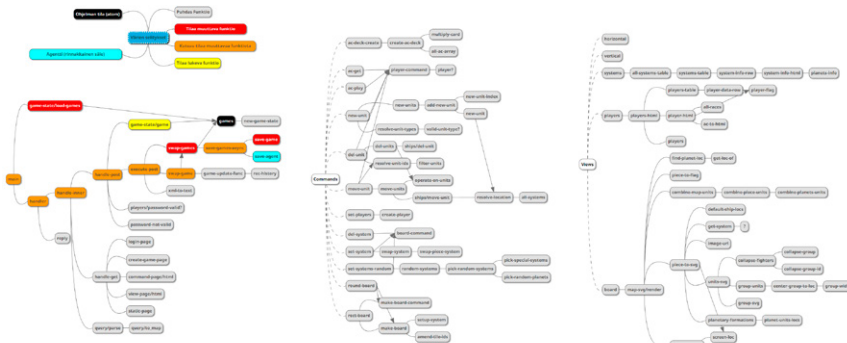
Toinen laajempi Clojure-projektini toteuttaa avaruusaiheista sotaa *Twilight Imperium* -henkisen monen pelaajan lautapelin muodossa: <https://github.com/rbrother/glory-of-empires>. Palvelinpuoli on tehty täysin Clojurella ja käyttöliittymä on HTML-web-sovellus. Web-käyttöliittymä lähettää Clojure-tietorakenteina pyyntöjä palvelimelle ja saa vastauksena HTML:ää tai JSON-viestejä.

Koodirivejä on noin 2000 ja funktioita noin 150. Kuvassa 1 näkyy kaikki funktiot ja niiden väliset kutsuriippuvuudet.

Ohjelman uusi pelitila on hakemisto. [Ks. listaus 7.] Kaikki ohjelman tila on yhdessä **atom**-tallennuspaikassa. Koska ohjelma tukee useita rinnakkaisia



Glory of Empires. Avaruussaiheinen heksapohjainen lautapeli. Pelitila on Clojure-tietorakenteena, ja siitä voi generoida erilaisia näkymiä. Karttaa rakennetaan ja aluksia liikutellaan komennoilla, jotka käsitellään palvelimella pelitilaa muuttavina Clojure-funktioina.



Kuva 1. Edustava otos funktioita Glory of Empires -pelistä. Funktiot jakautuvat kolmeen ryhmään: HTTP-pyyntöjä käsittelevät funktiot, pelikomentoja toteuttavat funktiot (*Command*) ja HTML/SVG-näkymiä luovat funktiot (*View*). Ohjelma noudattaa pitkälle edellä kuvattua arkkitehtuuria: tila on yhdessä atom-tyyppissä (musta), vain pieni joukko funktioita lukee tai muuttaa tilaa (punaiset) ja loput funktiot ovat puhtaita. Imperatiivisessa tyyliässä komentofunktiot muokkasivat globaalia tilaa ja näkymäfunktiot lukisivat globaalia tilaa. Funktionaalisessa tyyliässä molemmat ottavat pelitilan parametrina ja palauttavat siitä tuloksen – komentojen tapauksessa uuden pelitilan, näkymien tapauksessa HTML-koodia.

```
(def games (atom { :sandbox (new-game-state "") })))
```

Listaus 8.

```
(defn swap-game [ func game-id ]
  (swap-games (game-update-func func game-id)))

(defn- game-update-func [ inner-func game-id ]
  (fn [ games ] (update games game-id inner-func)))
```

Listaus 9.

```
(defn swap-games [ swap-func ] (save-games-async (swap! games swap-func)))
```

Listaus 10.

```
(def save-agent (agent nil))

(defn- save-games-async [ data ] (send-off save-agent save-game data))

(defn- save-game [ agent-state state ]
  (write-to-file game-file-path state))
```

Listaus 11.

pelejä, päätason tietorakenne on hakemisto, jossa avaimena on pelin nimi ja arvona pelin tila. Alkuarvona kokonaistilalle on yksi "sandbox"-niminen peli, jolla on tyhjä pelitila. [Ks. listaus 8.]

Pelitila vaihdetaan uuteen **swap-game**-funktiolla, joka ottaa parametreinaan pelitilaa päivittävän funktion ja päivitettävän pelin id:n. Se kutsuu **game-update-func**-funktiota, joka muodostaa koko ohjelma tilan päivittävän funktion yhden pelin päivittävästä funktiosta käyttämällä Clojuren **update**-funktiota. [Ks. listaus 9.]

Sitten **swap-games** kutsuu **swap-games**-funktiota, joka Clojuren **swap!**-funktiolla päivittää ohjelman tilan. Tämä on ohjelman ainoa sivuvaikutuksellinen kohta. [Ks. listaus 10.]

**Swap-games** myös kutsuu samalla **save-games-async**-funktiota, jolla pelitila tallennetaan turvallisesti taustasäikeessä **agent**-objektilla ja sen **send-off**-funktiolla hidastamatta muuta ohjelmaa. [Ks. listaus 11.]

Seuraavassa esimerkki näkymää tuottavasta piirtofunktiosta. Pelin kartta ja sen objektit tehdään SVG:nä. Tämä funktio tuottaa pelilaudan yhdessä ruudussa olevien alusten kuvat ryhmiteltynä yksikkötyypin mukaan. Clojuren makro `-->` mahdollistaa funktioiden kutsumisen sarjassa ilman suurta määrää sulkujia. Funktiossa aloitetaan **units**-listasta, ja sille kutsutaan järjestyksessä **collapse-fighters**, **group-units** ja **mapcat**. Funktionaalinen **mapcat** kohdistaa funktion listan alkioihin ja yhdistää tuloslistat yhdeksi. [Ks. listaus 12.]

Edellisen käyttämä **group-svg**-funktio käyttää funktionaalisessa ohjelmoinnissa tärkeää **rekursiota** (kutsuu itse itseään) tuottaakseen alusryhmän grafiikan kohtaan (x,y). Alukset ovat ryhmässä vierekkäin, eli joka rekursiolla **next-loc**-symboli saa arvon, jossa x-koordinaattia on kasvatettu aluksen leveyden verran. Clojuren **conj**-funktiolla rakennetaan pala palalta lista yksittäisen aluksen **svg**-funktiolla tehtävistä grafiikoista. Rekursio tarvitsee lopetusehdon, joka tässä funktiossa on se että alusryhmä on tyhjä. [Ks. listaus 13.]

Seuraavaksi esimerkki pelitilaa muuttavan funktion toteutuksesta. **Move-units**-funktio liikuttaa pelaajan aluksia ruudusta toiseen. Pelitila



```
(defn units-svg "Generates SVG for all ships in system"
  [ [ units group-locs-func ] ]
  (->> units
    (collapse-fighters)
    (group-units group-locs-func)
    (mapcat ships/group-svg)))
```

Listaus 12.

```
(defn group-svg [ [ group [ x y :as loc ] ] ]
  (if (empty? group) []
      (let [ { type :type count :count :as ship } (first group)
            next-loc [ (+ x (width ship) 1) y ] ]
        (conj (group-svg [ (rest group) next-loc ] )
              (svg ship loc))))))
```

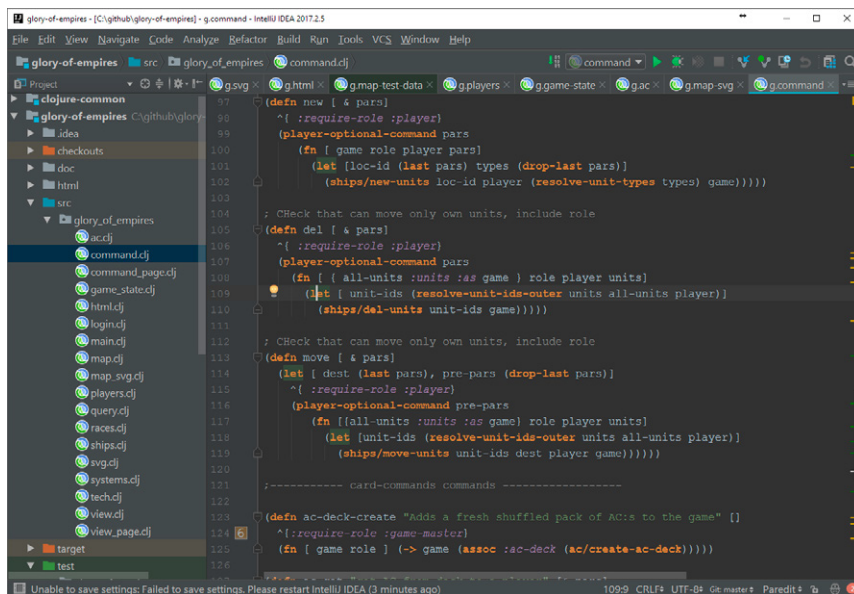
Listaus 13.

```
(defn move-units [ unit-ids loc-id game ]
  (let [ move-unit-to
        (fn [ units-map id ] (move-unit units-map id loc-id game)) ]
    (operate-on-units game move-unit-to unit-ids)))
```

Listaus 14.

```
(defn operate-on-units [ game units-fn unit-ids ]
  (update game :units (fn [units] (reduce units-fn units unit-ids))))
```

Listaus 15.



Cursive. Clojurelle on useita hyviä editoreita ja IDE-ympäristöjä. Itse käytän Intellij IDEA:n päälle tehtyä ilmaista Cursive-editoria, joka tukee monipuolisesti Clojuren editointia. Hyvä editori tukee muun muassa LISP-kielissä tärkeiden sulkusyntaksien kirjoitusta.

tulee **game**-parametrissa, ja funktio palauttaa uuden pelitilan. Funktiossa muodostetaan **fn**-lauseella paikallinen **move-unit-to**-funktio, joka käyttää hyväkseen isäntäfunktion parametrien **loc-id** ja **game** arvoja. Tällaista isäntäfunktion parametrien käyttöä paikallisessa funktiossa kutsutaan *sulkeumaksi* (engl. *closure*), ja se on myös tärkeä väline funktionaalisessa ohjelmoinnissa. Jopa niin että Clojure-kielen nimi tulee sa-

noista *closure* ja JVM. Lopuksi **move-unit-to**-funktio annetaan parametrina **operate-on-units**-funktioille, jota käyttävät myös yksiköiden lisäämiseen ja poistamiseen käytettävät funktiot, jolloin vältetään koodin toistolta. [Ks. listaus 14.]

**Operate-on-units**-funktiossa päivitetään **update**-funktioilla pelitilan **units**-attribuutti. Päivitykseen käytetään funktionaalista **reduce**-funktioita, joka ketjuttaa funktiokutsun listan alkiolle. [Ks. listaus 15.]

## Kohti funktionaalista tulevaisuutta?

Aina LISP:n syntymästä 1950-luvulla alkaen on aina aika ajoittain näyttänyt siltä, että funktionaalinen ohjelmointi on murtautumassa valtavirtaan. Imperatiiviset kielet ovat kuitenkin aina onnistuneet säilyttämään valta-asemansa.

Funktionaalinen ohjelmointi ei ole kuitenkaan mustavalkoinen joko-tai-ominaisuus tai vain funktionaalisilla kielillä mahdollinen malli. Mahdollisimman funktionaaliseen ohjelmointiin voi pyrkiä millä ohjelmointikielillä tahansa. Millä tahansa kielellä voidaan tehdä esimerkiksi puhdas funktio, joka ottaa kaksi lukua ja palauttaa niiden summan käyttämättä paikallisia muuttujia ja muuttamatta ohjelman tilaa. Myös globaalien muuttujien välttäminen on kuulunut ohjelmoinnissa suosittelaviin tapoihin jo olio-ohjelmoinnin alkupäivistä saakka.

Viime vuosina varsinaisten funktionaalisten kielten ja oliokielten välinen ero on myös kaventunut, kun oliokieliin on tuotu funktio-objektit ja korkeamman asteen funktiot. Tämä voi johtaa varsinaisten funktionaalisten kielten suosion lisääntymiseen, kun ohjelmoijat tottuvat funktionaaliin mekanismeihin, jolloin kynnys siirtyä funktionaaliseen ohjelmointiin madaltuu. Toisaalta se voi myös johtaa funktionaalisten kielten suosion vähenemiseen, kun tarve niille vähenee.

Funktionaalisen ohjelmoinnin suosiota tulee varmasti myös tulevaisuudessa rajoittamaan sen älyllinen haaste: varsinkin alussa saattaa olla vaikeaa keksiä, kuinka jonkin tietorakenteen päivittäminen tehdään funktionaalisen ohjelmoinnin periaatteiden mukaisesti. Globaalit muuttujat ja olioiden kenttien muuttaminen ovat kätevydessään houkuttelevia menetelmiä, ja niistä eroon pääseminen vaatii ajatus-työtä.

Proessoriydinten määrän noustessa paine tehdä helposti ja turvallisesti rinnakaistuvia ohjelmia kasvaa kuitenkin jatkuvasti, ja tähän funktionaalinen tyyli on paras ratkaisu. Sitä paitsi funktionaalinen ohjelmointi on hauskaa, kun siihen pääsee sisään ja data soljuu koodaamiesi eleganttien funktioiden läpi kuin keväturo. 🌱

# EI NÄIN!

## KAIKKEIN KURJIMMAT KOTELOT

Teksti: Mikko Heinonen

Kuvat: Mikko Heinonen ja Manu Pärssinen

*Koko CD-levy oli formaattina ongelmallinen kuluttajille annettavaksi, ja jokainen meistä lienee myös aikoinaan rikkonut niiden särkyvästä muovista valmistettuja koteloita. Mutta ei ole mitään niin huonoa, etteikö sitä saisi vielä huonommaksi.*

**K**onsolipelien pakkausten muotoja ovat sanelleet monet eri tekijät. Jo 1980-luvulla monien koneiden pelimoduuleista tehtiin hieman erimuotoisia ja -kokoisia, jotta saatiin estettyä harmaatuontia alueiden välillä. Jopa aivan samannäköisiä muovipalikoita pakattiin silti erilaisiin koteloihin, jotta omat pelit erottuivat kaupan hyllystä.

Aluksi pelien pääasiallinen pakkausmateriaali oli pahvi, joka luonnollisesti usein repeytyi, vettyi ja muuten vain katosi. Lopulta moduuleita oli jäljellä paljon enemmän kuin niiden koteloita, ja ”boxed” eli koteloa sisältävä peli nousi hyvin nopeasti arvoltaan korkeammaksi. Myöhemmin etenkin Sega alkoi pakata pelejään hieman kestävämpiin muovikoteloihin, kun taas Nintendo jatkoi pahvilinjalla aina Gamicubeen asti.

### Hopealevyn reunus

Saavuttaessa 1990-luvun alkuun CD-levy oli valloittanut äänentoistomarkkinat. Edulliselle muovikiekolle sai tallennettua niin hyvälaatuista ääntä, ettei tavallinen kuluttaja osannut parempaa kaivata. Kääntöpuolena oli toki se, että kustannussyistä levyn tietoja suojasi vain muutaman millin murto-osan paksuinen muovikerros. Näin levy oli myös helppo tuhota virheellisellä käytöllä, ja paikoilleen juuttuvan albumin mielipuolinen säksätys muodostui tutuksi etenkin niille, jotka kuljettivat levyjä mukanaan autossa tai treenikassissa. Alivaltioshteeri-radiohupailussa kuultiinkin aikoinaan sketsinä ”Rikki-näisten CD-levyjen toivekonsertti”.

Levyn naarmuuntumisherkkyyden lisäksi hermoja koeteltiin toisinaan myös

CD-levyjen kotelointi. Kotelossa käytetty polystyreeni oli jäykkää ja kantta piti paikoillaan vain kaksi ohutta saranaa. Ei tarvittu kovinkaan suurta voimankäyttöä, että ne sai murtumaan. Lisäriesaa tuottivat levyä paikallaan pitävät ”kynnet”, jotka oli helppo katkaista pitämällä sormea väärin levyä irrottaessaan. Kun keskiö oli näin saatu rikki, kotelossa vapaasti heiluva levy yleensä naarmutti itsensä ennen pitkää.

Puutteistaan huolimatta CD-levy oli niin edullinen tapa tallentaa paljon tietoa, että siitä seuraajineen muodostui pian myös pelikonsolien ensisijainen tallennusväline. Pelit olivat paljon kalliimpia kuin musiikkilevyt, joten voisi kuvitella, että ne olisi voinut myös pakata paremmin. Totuus oli kuitenkin toinen.

### Segoitamme tämänkin

CD-konsolipelien ensimmäistä sukupolvea edusti Sega Mega-CD, joka Yhdysvalloissa tunnettiin Sega CD:nä. Sen pelien pakkauksissa nähtiin ainakin neljää variaatiota: normaalia CD-koteloita (Japanissa), tupla-CD-koteloita (myös yhdelle levyille, etenkin Euroopassa), pahvikuorista CD-koteloita sekä erityisesti Amerikassa käytettyä tupla-korkeaa koteloita.

Näistä viimeinen ansaitsee tulla huomioiduksi erityisen huonoutensa ansiosta. Se ikään kuin moninkertaistaa alkuperäisen CD-kotelon ongelmat: helposti halkeavat isot pinnat ovat nyt vieläkin isompia, murtuvat pienet saranat suhteessa vieläkin pienempiä. Lisäkorkeus mahdollistaa toki isomman ohjekirjan asentamisen koteloon, mutta tämä on laiha lohtu, kun saranat katkeavat jo riittävän ilkeästi katseesta. Pelilevyn pysyminen paikallaan yritettiin varmistaa vaahtomuovin palasella, joka tipahti ulos joka kerta koteloita avattaessa ja katosi näin ollen takuuvarmasti hyvin pian.

Jostain syystä tätä mallia pidettiin sen verran hyvänä, että Sega käytti sitä myös seuraavassa CD-konsolissaan Saturnissa – ja myös Sony Playstationin ensimmäiset Amerikan-julkaisut oli pakattu samanlaiseen jättikoteloon. Onneksi sentään saranatappien kokoa muutettiin sen verran, että paljon yleisemmästä Saturn-kotelosta ei voi ottaa vaihtokantta Sega CD -koteloon.

### Vanhan mantereen vaikeudet

Euroopassa Sega ei sortunut Jenkkilän virheisiin, vaan Saturn-pelit saivat omanlaisensa koteloa. Valitettavasti vain se oli mahdollisesti jopa vielä huonompi kuin Atlantin toisella puo-







lella. Pahvirunko yritti pitää paikallaan kahta muovikotelon puolikasta, joiden sisällä pelilevyä puolestaan yritti kannatella heppoinen keskiö. Silloin, kun kotelon kiinnitys ei suoraan pettänyt, putosi levy usein kotelon sisään pyörimään. Pahvinen kansi rispaantui ja hintalapputarroja oli mahdotonta irrottaa siististi.

Sony taas oli siirtynyt USA:ssa alun hapuilun jälkeen pääosin normaalkokoisiin CD-koteloihin, joita Japanissa käytettiin alusta asti. Euroopassa kuitenkin haluttiin keksiä pyörä uudelleen ja Playstation-pelit saivat hieman CD-koteloa paksumman muovilaatikon. Taustalla lienee ollut se, että monikielisellä mantereellamme pelin mukaan piti pakata paljon paksumpi ohjekirja kuin virallisesti yksikielisissä USA:ssa ja Japanissa.

EU-mallinen Playstation-kotelo ei ollut aivan kelvoton tekele, mutta ei juuri CD-koteloa parempikaan. Materiaali oli edelleen liian jäykkää ja haurasta, joten saranat napsahtelivat poikki ja kannet halkesivat tuon tuostakin. Lisäksi peruskotelosta oli liikkeellä myös muutaman kustantajan omaa versiota, joihin oli taas kerran vaikeaa löytää varaosia mistään, kun väistämättömän lopulta tapahtui.

### Unelmakonsolin painajaiskotelo

Kaikesta tästä huolimatta se konsoli, jonka pelikoteloita olen itse eniten rikkonut, on selkeällä erolla muihin Sega Dreamcastin PAL-versio. Sen kotelossa onnistuttiin, ihme kyllä, yhdistämään lähes kaikkien edellä mainitsemieni koteloiden huonot puolet.

Dreamcast-pelit pakattiin Euroopassa aina koteloihin, joissa oli paikka kahdelle levyille. Käytännössä tästä onkin kuluttajalle hyötyä, sillä levyä paikallaan pitävät keskiötäpit ovat aivan uskomattoman heikkoja. Ne saa katkaistua, vaikka olisi kuinka varovainen. Tuplauksen ansiosta virheen saa sentään tehdä kerran ja levy pysyy edelleen paikallaan kotelossa, kunhan sen vaihtaa toiselle puolelle.

Saranoiden ongelmaa onnistuttiin pahentamaan muotoilemalla etu- ja takakansi niin, että muovi on ohuimmillaan juuri siinä, mihin taivutus kohdistuu, kun melko tiukasti sulkeutuvaa koteloa yrittää avata. Vähänkin huolimattomalla käsittelyllä on vain ajan kysymys, milloin sarana murtuu vähintään toiselta puolelta. Kuin kermana kakun päälle Dreamcastin alle 250 PAL-pelin valikoimaan on saatu myös mahdollitettua useampia hieman erilaisia kotelotyyppisiä, mitä täydellistä (ja ehjää) PAL-kokoelmaa kasaavat harrastajat ovatkin runsaasti kiitelleet. Japanissa ja USA:ssa käytössä olivat tavalliset CD-kotelot.

### Ennen ei ollut paremmin

Pelikoteloiden särkyminen on niitä harvoja ongelmia, joiden voi sanoa poistuneen lähes kokonaan tai ainakin rajusti vähentyneen. Playstation 2:sta alkaen pelit on pakattu huomattavasti pehmeämpiin DVD-koteloihin – samoihin kuin elokuvatkin. Vaikka tästä säännöstä on hieman valmistajakoh-

taisia poikkeuksia etenkin muodon suhteen, materiaali on kaikilla samankaltaista, eivätkä saranat enää katkeile tai kannet halkea läheskään samaan tahtiin kuin ennen. Lisäksi jonkinlainen standardointi varmistaa, että ehjiä kuoria voi vaihdella peleistä toisiin. Puhumattakaan sitten siitä, että nykyisin yhä useampi peli myydään suoraan digitaalisena.

Näin ollen rikkinaisten kuorien kanssa ovat tekemisissä nykyisin lähinnä vanhojen pelien keräilijät. Varsinkin vanhempien CD-pelien lähettäminen postissa on edelleen käänteistä lottoa: huolellisestikin pakattu 90-luvun pelikotelo saa helposti kuljetuksessa siipeensä, pelkässä kuplamuovikuoressa taas rikkoutumisesta voi olla lähes varma. Ja sitten taas kerran pikaliimataan. 🐛





# Digiarkeologiaa Kasettilamereiden seurassa **2.0**

*Skrollin ensimmäisessä numerossa esittäytyi digiarkeologiryhmä Kasettilamerit. Tuolloin ryhmä haki vielä muotoaan, mutta toiminta on sittemmin sekä järjestäytynyt että laajentunut huomattavasti.*

Teksti: Aki Sivula

Kuvat: Tommi Lempinen, Aki Sivula, Jorma Oksanen, Jukka O. Kauppinen, Atte Haapalahti

**K**asettilamerien toiminta haki Skrollin alkuaikoina (ks. Skrolli 2013.1) vielä osin muotoaan, mutta kuluneen neljän vuoden aikana ryhmän toiminta on kuitenkin kehittynyt tuntuvasti. Onpa keskusteltu myös virallisen yhdistyksen perustamisesta. Vanhat kasetit ja levyt eivät ole edelleenkaan monen mielestä mielenkiintoisia, mutta sisältö sentään kiinnostaa melko laajaa yleisöä nostalgian ja taannaisen retrobuumin kautta.

Myös Suomen pelimuseo on tehnyt asiaa uudella tavalla kiinnostavaksi, pääseehän museossa pelaamaan aidoilla laitteilla ja saa samalla pelien historiasta lyhyen oppimäärän.

Arkistointi vaikuttaa monien mielestä työmaalta, jossa ”kaikki on jo tehty”. Kasettilamerien mukaan totuus on kuitenkin toinen, sillä työtä riittää edelleen tehtäväksi. Arkistointimatonta materiaalia löytyy edelleen paljon, sillä toiminta on laajentunut huomattavasti alkuperäisestään. Niinpä Skrolli sukeltaa seuraavaksi

Kasettilamerien **Aki Sivulan** silmin ryhmän nykytilaan, löydöksiin, matkalla törmätyihin oimutuisuuksiin ja ryhmäläisten aikaansaannoksiin.

## 3”–8” ja nauhat päälle

Kasettilamerit on samanhenkisesti ystäväporukasta koostuva vuonna 2011 perustettu magneettisen ja optisen median arkistointiin erikoistunut



Lamerointia on tehty myös julkisesti yleisön edessä. Vanhan datan pelastaminen jälkipolville kiinnostaa, niin outoa kuin se onkin. Kuva: Atte Haapalahti / Assembly Organizing.



digiarkelogiryhmä. Ryhmä rakastaa Commodorea mutta tallentaa nykyisin minkä tahansa alustan medioita aina kaseteista disketteihin ja kovalevyistä nauha-asemiin. Kolme vuotta sitten ryhmä pystyi tallentamaan vain Commodoren Turbo tape -kasetteja sekä 5¼ ja 3½ tuuman levyjä, mutta sen jälkeen on tallennettu suorastaan omituisiakin formaatteja. Vai miltä kuulostaa Nintendon disketti, joka toimii kuitenkin kuin kasetti? Entä saksalaisessa museossa toimiva Siemensin teleprinteri, joka käytti custom-formaattiin tehtyä lisenssidiskettiä?

Nykyisin tiimi ei enää tunnusta mahdollisia formaatteja. Toki 70-luvun suuria nauha-asemia ei meiltäkään löydy, mutta olemme kiinnostuneita myös niistä. Tietotaitoa kasvatetaan koko ajan hankkimalla puuttuvaa rautaa ja tekemällä jatkuvasti lisää. Tuki löytyy jo ainakin 3, 3½, 5¼ ja 8 tuuman disketeille sekä useimmille ka-

seteille. Kovalevyt, C-kasetit, ROM, EPROM, Flash ja VHS (myös Amigan *Video Backup System*) ovat nekin tuettujen listalla.

En tiedä kuinka monta kymmentä kertaa olen kuullut väitettävän, että ”kaikkihan on jo arkistoitu ja tallessa”. Commodore 64:n ja Amigan osalta tilanne on jo aidosti hyvä ja paranee koko ajan, mutta se ei suinkaan ole koko totuus. Erilaisia crack-versioita löytyy kyllä, mutta crackitkin on tehty aina alkuperäisestä versiosta. Edelleen löytyy aiemmin tuntemattomia demoja ja introja, puhumattakaan entisaikojen Suomi BBS:ien varmuuskopioista. Vastaan on tullut useita pelejä, joista ei ole yhtään crack-versiota tai versio on yksinkertaisesti rikki tai virheellisesti murrettu. Joistain on liikkunut pelkästään kasettiversio ja toisista diskettiversio. Esimerkiksi Apple II:n alkuperäisiä julkaisuja ei ole arkistoitu nimeksikään, mutta sil-



Nasse-Setä Oy ja Amersoft jäivät ilman terveisiä.

ti niitä ”arkistoidaan” crack-versioina Internet Archiveen. Hyvää työtä sekin, mutta toivoisin myös 1:1 vastaavia jäljennöksiä Apple II:n osalta. Sopivia keräilijöitä ei ole kuitenkaan ilmoittautunut vapaaehtoisuuteen.

## Itse tekemällä saat parempaa

Ammoisina 1980-luvun vuosina harva mietti minne heidän tekemänsä pelit, tekstit ja muut tuotokset päätyisivät.

## Löydöksiä vuosien varrelta

*Lord of the Ringsin* C64-kasettiversio näyttää latauskuva, mutta ei itse asiassa lataa mitään kuvan ilmestymisen jälkeen kasettiaseman 30 seuraavaan laskurikierrokseen. Latauskuvan ilmestyessä voi siis huoletta kelata kasettia eteenpäin päästäkseen pelaamaan nopeammin. Havainnon teki **Jorma Oksanen** peliä dumpatessaan.

*James Bond View to A Killin* alkuperäisversion kolmannessa kentässä on karttabugi, joka johtuu siitä, että tiedoston tallennus on ylikirjoittanut muutaman tavun Basic-muistin yläpäässä. Muutenkin pelin koodissa on hutiloinnin merkkejä. Virhe löytyi pelin karttojen luonnin ja koodin tutkinnan yhteydessä. Hakerointia harrasti Jorma Oksanen.

Lähes kaikista Infocomin ja Microprosen peleistä on julkaistu useita eri versioita, joissa on bugikorjausten lisäksi huomattavia eroja. Dumpeja tutkitaan ja vertaillaan jatkuvasti, joten nämä löytyivät rutiinityön yhteydessä sekä keskustellen Software Preservation Societyn kollegojen kanssa.

CRL:n peleistä löytyy hienommilla grafiikoilla toteutettuja C128-versioita. Esimerkiksi *The Rocky Horror Show* saatiin saksalaisen avustajan toimittamien dumpien joukosta. Strattonin mahdollinen C128-versio on edelleen etsinnässä.

Hieman yllättäen Commodore 64:n *Top Gunista* on julkaistu useita eri versioita.

Nämä ovat tulleet vastaan dumpien tarkastuksen yhteydessä.

Kasetilla julkaistua DK'Tronicsin *Invasionsia* ei ollut olemassa edes crack-versiona, eikä peliä edes tiedetty olevan olemassa. Nyt kasetista on 1:1 jäljennös. Jorma Oksanen löysi pelin sattumalta normaalien kasettien dumpauksen lomassa.

Oceanin *We Are the Champions* -kasettikokoelman *IK+* on suora muistidumpi, joka kuitenkin ladataan Freeload-järjestelmän avulla. Luultavasti alkuperäinen master-kasetti on ollut hukassa tai muistidumpin käyttöön on päädytty perinteisen kiireen tähden. Tämä havaittiin dumpien tutkinnan yhteydessä.

*J.R.R. Tolkien's War in the Middle Earth* -pelistä löytyi SPS:n arkistoinnin yhteydessä aiemmin tuntematon kahden disketin versio, jossa oli hienommat grafiikat kuin tutussa yhden levyn versiossa. Tästä versiosta ei myöskään ole crack-versiota C64-pelejä keräävillä csdb.dk- tai Gamebase 64 -verkkosivuilla.

Australiasta on löydetty lukuisia paikallisesti julkaistuja pelejä, joista ei ole olemassa tunnettuja crack-versioita. Kurioositeettiä Imagine ei koskaan julkaissut *Mikiä* Euroopassa disketeillä, vaan ainoastaan kasetilla. Ozisoft Australiassa julkaisi siitä halpiversion – kaikki muut versiot ovat olleet kokoelmilla. C64:n kasetista pyydetään nykyisin jopa 500 euroa. Suurin osa puuttu-

vista Commodore 64:n nimikkeistä löytyi edelleen Yhdysvalloista ja Australiasta. EU:n osalta C64:n tilanne on jo varsin hyvä.

*Chuck Rockin* C64-versio julkaistiin vain Italiassa. Vuosia luultiin, ettei Core Design julkaisut peliä ollenkaan. Sen etsintä oli melkoinen operaatio, ja ehjä diskettiversio löytyi sattumalta eräältä suomalaiselta keräilijältä. Kasettiversiosta taisin köyhtyä useamman satasen. Mikäli asia kiinnostaa, niin sitä on käsitelty melko laajasti Games That Weren't 64 -sivustolla. Sieltä voi myös ladata läpipelattavan alkuperäisversion.

Saksalainen SPS:n kollega löysi Saksan eBaysta *Giana Sistersin* uudet ja käyttämättömät Amiga-, Atari ST- ja Commodore 64 -versiot. Tähän mennessä kaikissa aiemmin tunnetuissa versioissa oli tallennettu high score -listoille vähintään yksi tulos, minkä takia ne eivät kelpaa alkuperäisten versioiden arkistointiin. Myyjä oli entinen THQ/Softgoldin kehittäjä. Pelit omistanut henkilö toi mediat Saksassa paikan päälle henkilökohtaisesti, ja vihdoin jo kuusi vuotta etsitty Commodore 64 -versio saatiin talteen.

Assembly Summer 2017 -tapahtumassa pelastettiin harvinainen Jari Paulinin tekemä suomipeli *Mr. Seek*. Tapahtumassa Aki Sivula tallensi Spectravideo SV-318/328- ja MSX-versiot. Dumpin sielunelämän selvitti sekä konvertoi Tommi Lempinen.

Sinus-scrolleri on tehty animoimalla fonttia ja niin edelleen, mutta minusta sekin osoittaa luovaa ongelman ratkaisua. Ei sen väliä miten se on tehty, kunhan lopputulos on hieno. Taidot kyllä kehittyvät tekemällä.

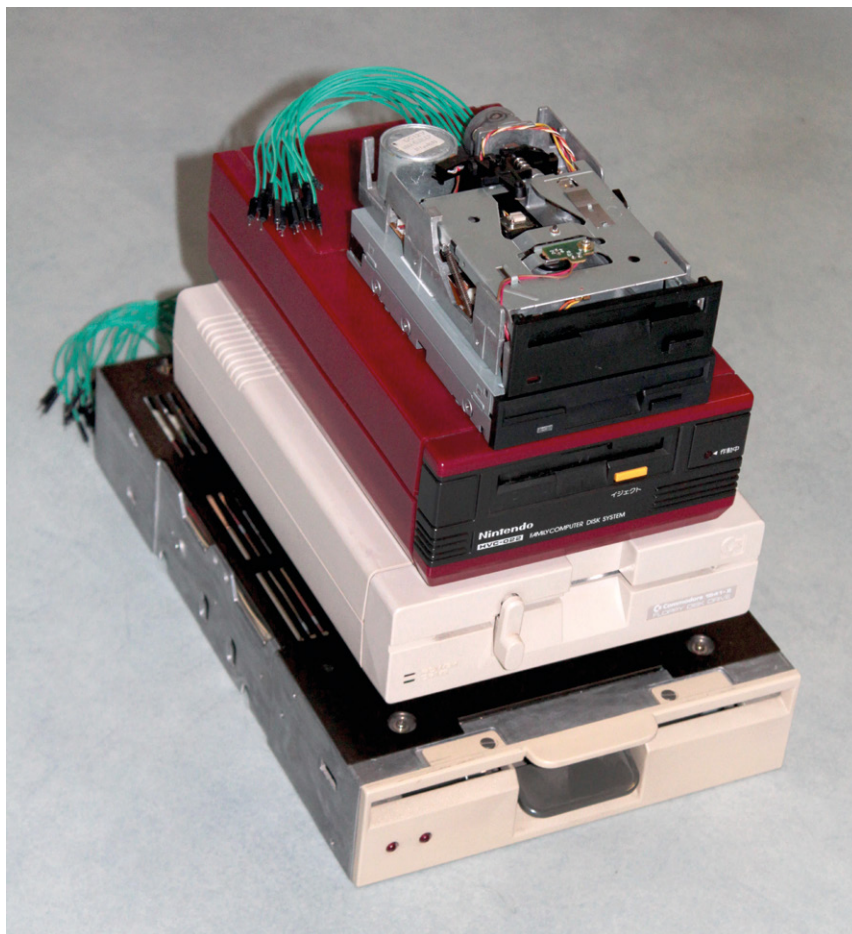
Jotkut omatekoiset demomakerilla tehdyt demot ja pelit ovat tahattoman koomisia. Pelihahmo saattaa hypätä ilmaan ja lentää ruudusta ulos, tuloksena sovelluksen kaatuminen. Omatekoisissa seikkailupeleissä on joskus epäselvää mitä pitäisi tehdä tai komentoja tutkiva parseri on hieman kömpelö. Hirviö syö sinut ilman sen kummempaa syytä ja niin edelleen.

Toteuttajien joukossa on ollut huipulajahjakkuuksia, tavallisia poikia ja tyttöjä, Viznutin hauskojen PC-lamerien tyyppisiä wannabe-skeneriä, pornofriikkejä kuin myös taiteilijan alkuja ja koodareita.

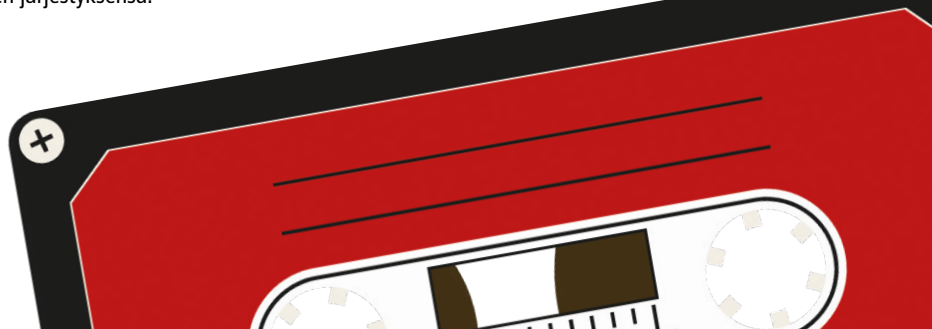
Kotitekoisilla medioilla törmää mitä omituisimpiin löydöksiin. Basic-kielen, piirto-ohjelmien ja musiikintekosoftien avulla monet harrastajat pääsivät tuolloin ensimmäistä kertaa uudennaiselle digitaaliselle näyttämölle. Syntyi Deluxe Paintilla piirrettyjä kuvia, Noise trackerilla tehtyä musiikkia, basic-demoja, ohjelmia ja pelejä. Joskus ne olivat pelkkää hauskanpitoa ilman selvyttä alkuperäisestä tarkoituksesta. Hurjimmat onnistuivat loihtimaan assemblerilla suoranaista taikuutta.

Kunnon lamerihengessä monia crack-pelejä on myös surutta muuteltu omiin nimiin heksaeditoreiden ja erilaisten basic-kötöstysten avulla. Kuinkahan monta kopiota alkuperäisestä C64:n Hawkeystäkin tehtiin Copy-Q:lla ja muuteltiin vain heksaeditorilla ensimmäisen valikon tekstit? Allekirjoittanutkin uskaltaa tunnustaa tehneensä kyseisestä pelistä oman 'julkaisunsa' 30 vuotta sitten, mutta kaveri ylikirjoitti ne omilla nimillään ennen seuraavalle kopiointia.

Turbo tape -kasettien nimet ovat toisinaan varsinainen aarreaitta, vai miltä kuulostavat nimikkeet: TACK&FIELD, ARTIGALLERY DUEL, BLOD'N GUTS, BURNINIG RUBBERS, GUEST FOR TIRES, CIANA SISTERS, EXPLONING FISH, FRADIC FREDDIE, FUCK APE, GALAXSY, JUNKLE HUNT, INSPEKTOR GADGET, HUND BACK II, JOKI KARHU, ROPOTHON 2084, GOPY-Q II, RADAR-



Vanhan datan pelastaminen vaatii usein poikkeuksellisia välineitä. Laitteiden sekamelska on kuitenkin vain pintapuolista, sillä teknisessä kaaoksessa piilee oma looginen järjestyksensä.







Suomipeliklassikko *Painterboyn* alkuperäistä materiaalia löytyi Tikkurilan varastosta sekä lameroitavaksi että Suomen pelimuseoon toimitettavaksi.



*Little Knight Arthur*- ja *Uuno Turhapuro muuttaa maalle*-pelit tehnyt Pasi Hytönen näki 30 vuotta vanhan Arthur-pelinsä heräävän eloon kasettilamerien käsissä.

RATARACE, JUMPPI UKKO, SATR RAIDERS II, SUISE STRIP, SHUTLE SIM tai SAVUNAHKA JACK?

Erilaisten kopiopelien joukossa on toki oman aikansa *Hello world* -virityksiä, kuten TIINAN EKA OHJEL. Tarina ei tosin kerro, päätykö Tiina IT-alan tehtäviin. Jonkin verran löytyy myös varsin hienoja, jo unohduksiin painuneita sovelluksia. Monet tuntemattomat demot ja intro ovat myös todella hienoja. Kannattaa siis selata myös tuntemattomien ryhmien tuoksia, sillä voitte yllättyä positiivisesti.

5¼" lerppujen suojataskut ovat olleet myös varsinainen aarreaitta, niistä on löytynyt paljon todella näyttävää taidetta. Luultavasti moni alalle päätenyt graafikko on noita piirrellyt ja käden jälkeä ei voi kuin ihaila. Uskonnolliset yhteisöt ovat niin ikään hyödyntäneet tietoisesti nuorison innostusta tietokoneisiin, sillä monille kopioituille

levykkeille on lisätty uskonnollisia kirjoituksia. Esimerkiksi monille Amigan *Sound Trackerin* sample-korpuille on lisätty kirjoituksia Vartiotorinista. Ohjeita rikoksien ja erilaisten räjähteiden tekemiseen liikkui myös varsin varhain, ainakin demoryhmä *Dual Crew* kunnostautui tällä varsin kyseenalaisella alueella. Tietotekniikan villi länsi oli siis todistettavasti olemassa myös 30 vuotta sitten, niin hyvien kuin pahojen lieveilmiöiden kera.

### Arkeologiaa väärillä välineillä

CSDB:ssä (Commodore 64 Scene Database) joillain C64:n D64-levyimageilla havaitut virheet johtuvat usein siitä, että ne on alkujaan luettu arkistointikelvottomalla laitteistolla, kuten perinteisellä Commodore 1541 -levy-aseamalla tai sen kloonilla.

Nykyisin nämä virheet pystytään havaitsemaan. Yksi esimerkki huo-

nosti arkistoidusta piraattiversiosta on CSDB:stä löytyvä *P-47 Thunderbolt* -peli (Sharks crack). Peli on salakavalaasti rikki bittisiirtymän tähden, eikä tätä voi havaita levyn tallennuksen yhteydessä vaikkapa Zoomfloppyn tai jonkin siirtokaapelin avulla levyjä tallennettaessa. Kentässä 8 on kaksi tavua rikki bittisiirtymän tähden, ja kohta näyttäisi olevan varsinaista pelin koodia. Näin ollen peli ei välttämättä ole läpäistävässä. Niinpä sen kaikki kopiot kannattaisi tallentaa kunnollisen laitteiston, kuten Kryofluxin ja sen sovelluksen avulla. Tällä tapaa omatkin kotitekoiset tuotokset tallentuisivat luotettavasti.

Samasta tietokannasta löytyy paljon muitakin rikkinäisiä D64-imageja, jotka ovat näennäisesti kunnossa. Valitettavasti virheet ovat voineet aiheutua jo 80-/90-luvulla, jolloin kaikki kopioitiin koneen omilla levyasemilla. Commodoren levyillä käytetty DOS-sektorien tarkastussumma on nimittäin heikko datan eheyden tarkastamisen kannalta. XOR:iin perustuva tarkastussumma voi merkitä sektorin ehjäksi, vaikka se ei itse asiassa sitä ole. Tämä on sinänsä vanha tieto, mutta aiheutti aikoinaan paljon omituisia virhetilanteita, kun kaverille kopioitu disketti vaikutti ehjältä mutta silti kopio ei jostain kohtaa toiminutkaan. Periaatteessa ainoa tapa tarkastaa vanhoja 1541:llä tallennettujen levyjen tai D64-tiedostojen virheitä on verrata niitä tunnettuun ehjään dataan, eli vaikkapa verrata eri ryhmien crack-versioita keskenään. Pelien osalta eräs vaihtoehto on verrata sitä Kryofluxilla alkuperäisestä pelistä tehtyyn D64-tiedostoon. Hakkerin taidoista ei siis ole haittaa, jos syvemmälle haluaa mennä.

### Alkuperäisissäkin yllätyksiä

System 3:n *The Last Ninja II* (C64) on erikoinen tapaus. Pelin alkuperäinen EU-versio on itse asiassa crack-ryhmä Ikarin julkaisema versio, ja Ikarin nimi löytyy kaikista EU-versiosta levyn raakadataa tutkimalla. Oceanin *100% Dynamite* -kokoelmalla on sama crack-versio, mutta lisäksi The Mansi-on -kenttä on rikki.

Crack-version käyttämiseen päädyttiin mahdollisesti master-levyn luonnin yhteydessä ajan säästämiseksi. Niinpä levyiltä poistettiin ainoastaan Ikarin crack-intro. Tuolloin Oceanilla

masteroinnista vastannut Peter Hughes ei myöhemmin kysyttäessä enää muistanut tapausta, mutta System 3:lla on varmasti ollut peli lähdekoodeineen ja se on vain vuotanut jossain vaiheessa. Levyn masteroinnin yhteydessä on ollut helpompaa ottaa valmiiksi toimivat binäärit, ja siinä se. Samaa tempua on hyödynnetty nykyaikanakin, joskin tämä lienee yksi varhaisimmista tapauksista, jossa piraattiversio laitetaan myyntiin alkuperäisen sijaan. Viestintää pelejä murtavien ja pelejä valmistavien yritysten välillä tämäkin.

## Uuno, Painterboy, Moonfall ja monet muut

Pikkupoikana ei tullut mieleenkään, että tulisimme joskus tapaamaan kuuluisien demojen ja pelien kehittäjiä. Lapsuudessa näitä taideteoksiksikin verrattavia tuotteita kehittäneet henkilöt tuntuivat suorastaan mystisiltä yliihmisiltä. Nyt monia heistä tavanneena voi todeta, että ihmisiä hekin ovat tietoineen, taitoineen ja puutteineen. Taustalta on löytynyt oikeastaan aina palava into tekemistä kohtaan. Se lienee myös heidän menestyksensä salaisuus sillä menestys tulee harvoin ilman kovaa työtä.

Tähän mennessä moni demo, suomipeli ja sovellus on saatu talteen sitkeän yrittämisen, harrastajien tuen ja osin myös suurehkon taloudellisen panostuksen ansiosta. *Uuno Turhapuron* arkistointi Mediamuseo Rupriikissa oli tilaisuus sinänsä, ja **Jukka O. Kaupisen** ansiosta tilaisuudesta tallentui Checkpoint TV:hen videokuvaa ja muutamia haastatteluja.

*Painterboyn* arkistointi oli Assembly Summer 2015 -tapahtumassa julkinen



Vanhon kasettien ja levyjen tallentaminen on tieteellisen tarkkaa työtä.

ja lamereille merkittävä tilaisuus. Tuoloin Tikkurilan edustaja **Olli Ruhanen** toimitti alkuperäiset mediat paikan päälle tiimin arkistoitavaksi. Pelin levy- ja kasettiversiot sekä mainoskäyttöön tehty VHS tallennettiin yleisön silmien alla. Paikan päälle saapui myös *Painterboyn* aikanaan kehittänyt **Teijo Pellinen** lapsineen kertomaan taustoja pelistä ja sen kehitystyöstä. Erityiskiitos sitkeästä salapoliisityöstä kuuluu **Janne Rautiaiselle**, jota ilman tämän aikansa timantin tallennus ei olisi ollut mahdollista. Janne on ansioitunut myös vanhojen lehtien ja mainoksien skannauksessa sekä lupien hankkimisessa niiden julkaisuun netissä.

Ryhmämme hallussa on ollut jo vuosien ajan **Jukka Tapanimäen** alkuuperäiset C64:n ja PC:n kehityslevyt, jotka hänen äitinsä oli laittanut koneineen päivineen myyntiin paikallisen Siwan ilmoitustaululla. Levyt päättyivät meillemme ainoastaan onnekkaan sattuman vuoksi, koska Jarkko Lehden tuttava oli saanut koneen levyineen haltuunsa.

Levyiltä löytyi muunmuassa *Zamzaran* varhainen, Hewsonin alkujaan hylkäämä versio. Niinpä Tapanimäki koodasi pelin osittain uudelleen reilussa kuukaudessa Hewsonilta saadun palautteen perusteella. Nähtävästi Jukan kipakka suhtautuminen asioihin näkyi **Andrew Hewsonille**, sillä Jukasta tehty karikatyyri löytyy *Netherworldin* takakannesta. Ilmeisesti sama versio on ollut aiemmin levityksessä crack-versiona, joka on

taasen vuotanut aikanaan Hewsonilta.

Tapanimäen viimeiseksi jääneen pelin, *Moonfallin*, lähdekoodit löytyivät yhdeltä vaurioituneelta PC-lerpulta, jota yritettiin tallentaa useita eri kertoja. Tapanimäki nimittäin kehitti pelin Intelin prosessorilla varustetulla PC:llä ja käänsi koodit ristiinkääntäjällä. Tuloksena oli Commodoren 6502:lle sopivaa binääriä, jota pystyi suoraan testaamaan PC:n perään kaapeilla kytketyllä C64:llä. Levyllä oli kuitenkin virheitä, eikä lähdekoodeja saatu ehjänä ulos, mutta lopulta sitkeä yrittäminen palkittiin, ja koodit saatiin täydellisinä talteen. Tapanimäen ystävä **Esa Lyttinen** otti koodit talteen ja sanoi tekevänsä pelistä päivitetyn version.

Muita virstanpylväitä ovat olleet *Nero 2000* -tietovisapelin kehitystiimin löytyminen ja heidän materiaalinsa tallentaminen, puhumattakaan monista muista suomipeleistä. Toinen mainittava löydös on aiemmin tuntematon, innokkaan kalastajan suomipeli nimeltään *Ahvenjahti*. Peli on taitavasti toteutettu lähestulkoon julkaisukuntoon ja sitä voidaan pitää eräänlaisena PC:lle julkaistun *Pro Pilkin* esiasteena. Nähtävästi sopivan jakelijan löytäminen on estänyt pelin kaupallisen julkaisun, sillä pelin tekijä oli kirjoittanut myös käyttöohjeet ja löytyipä aineistosta jäljellä olevien bugien listakin. Tarkoituksena olisi saada peli julkaisuun hieman myöhemmin, joten salapoliisityö ja restaurointi jatkuu.

Kasettilamerit osallistuu myös muihin projekteihin, kuten The Old School Emulation Centerin tukemiseen. TOSEC on kaikkiin vanhoihin tietokoneisiin ja konsoleihin erikois-

Huora-intro ja muuta mukavaa.





tunut mammuttikanta, johon tallennetaan kaikkea mahdollista softaa, firmware-tiedostoja ja resursseja, jota voi ajatella tuhansien ja taas tuhansien sellaisenaan tallennettujen diskettien työssijana. Keskukseen on toimitettu muun muassa Splatterheadin 16 gigatavun kokoinen Amigan CD-kokoelma, joka tallennettiin 80- ja 90-luvun aikana. Aikanaan Scoopexin riveissä toiminut Splatterhead onnistuikin keräämään melkoisen määrän myös suomalaisissa BBS:issä liikkuneita tekstejä ja muuta mielenkiintoista. Kokoelmasta löytyi runsain mitoin Amigalle tehtyjä Suomi-julkaisuja, joita ei ole juuri muualla näkynyt.

## Commodoren salassa pidetyt kehitysversiot

Vierailut erilaisissa tapahtumissa ja demopartyillä ovat osaltaan tärkeä osa digiarkeologiaa. Tommi Lempienen avusti tallentamalla Revision 2017 -partyillä noin viidensadan ROM-piirin erää. Mukana oli niin julkaisematta jääneen Commodore 65:n prototyypin ROMEja, kuin muitakin varsin uniikkeja harvinaisuuksia historiasta. Näistä saattaa aikanaan löytyä emulaattoreiden kehittämisen kannalta uutta ja merkityksellistä tietoa.

Kyseiset piirit löytyivät tietävästi Saksasta entiseen Commodore-huoltoon yhteydessä olleen henkilön vinttiltä. Piirit ovat todennäköisesti peräisin Braunschweigin tehtaalta Saksasta, jossa Commodore tuotti suurimman osan Euroopan alueelle valmistetuista tuotteista. Eräs mukana olleista henkilöistä käsitteli tarkasti dokumentoiden osan kokoelmasta, joka sisälsi Commodoren piirejä, C65-diagnostiikkaa ja tietokoneiden kehittelyyn ja testaukseen käytettyjä ohjelmistoja. Mielenkiintoisista piireistä voisi mainita C65 DMAGIC REVISION 02:n ja kaksi DAPHNE / DENISE 8373 -piiriin prototyyppiä. Loput olivat erilaisia ROMEja Amigalle, Commodore PC:lle sekä muun muassa kovalevyille. Näitä on mahdollisesti diagnosoitu erilaisiin laajennuksiin yhteensopiviksi, koska keskustelujen perusteella kyseinen laitos valmisti myös muille kotimikroille laajennuksia.

Sanomattakin on selvää, että parvityaika meni tällä kertaa lähes täysin lameroinnin merkeissä ja aivan pikutarkaa työtä ei näiden muiden RO-

Mien osalta ehditty tekemään. ROMit on sittemmin toimitettu Zimmers.net-arkistoon ja muutenkin saatettu avoimesti emulaattoreiden kehittäjille tutkittaviksi. Tallennuksesta tehtiin jopa Revisionin Wild compoon sijalle viisi yltänyt ”Commodore live preservation project”. Video löytyy osoitteesta [www.pouet.net/prod.php?which=69733](http://www.pouet.net/prod.php?which=69733).

## The black hole of select elite

Skrollin ensimmäisessä numerossa olleesta artikkelista tuli jonkin verran palautetta, jopa vuosia sen jälkeen. Useimmat palautteista ovat olleet positiivisia, mutta moni purnasi myös siitä, miksi emme laita töidemme tuloksia yleiseen jakeluun verkkoon. Eräs lukija kommentoi, ettei koko lamerointi ole kovin kiinnostavaa.

Olen samaa mieltä. Harrastuksena digiarkeologia on varsin autistinen, mutta toisaalta se on yllättävän sosiaalista puuhaa. Ilman sosiaalisia ympyröitä tätä ei varmasti jaksaisi kukaan. Työ on vapaaehtoistyötä, emmekä saa tästä palkkaa tai korvausta. Totta kai teemme parhaamme, jotta työmme hedelmät jaetaan harrastajille. Jos tuot meille itse omat mediasiat, saat tietenkin kaiken tallennetun datan itsellesi riippumatta siitä, ovatko ne kotitekoisia vai alkuperäisiä nimikkeitä.

EU:n lainsäädäntö kuitenkin vaikeuttaa työtämme, eikä alkuperäisten levyjen digitaalisten kopioiden jakaminen onnistu suoraan. Jarkko Lehti ylläpitää Commodore 64:n demoskene-pakettia, jonka voi ladata Skrollin sivuilta. The Commodore 64 Scene Databaseen lisätään löydöksiämme sitä mukaa, kun muut harrastajat ehtivät niitä kantaan lisäämään.

Kannustaisin vapaaehtoisia liittymään mukaan toimintaan, sillä työtä riittää kyllä ja kaikki materiaali on aina tervetullutta. Muutenkin etsimme uusia jäseniä niin skannauksen kuin magneettisen median tallennukseen, joten rohkeasti vain kysymään jos lamerointi on sinulle se juttu. Kerromme kyllä toiminnastamme lisää ja siitä, mitä jäseneltä odotamme. Software Preservation Societyltä saat tiketin lähettämällä pelistä tai ohjelmasta identtisen jäljennöksen, jos pystyt todistamaan, että sinulla on alkuperäinen toimivana tai toimimattomana tai sinulla on muuten oikeudet kyseiseen nimikkeeseen. Se, mitä harrastajat tekevät saamallaan

## Lameritkin ovat ihmisiä

Kasettilamerointi on ennen kaikkea sosiaalista puuhaa, jota lamerit tekevät harrastuksenaan. Millaisia lamerit ovat ihmisinä ja mitä he lameroidessaan puuhastelevat? Lue lisää lamereista jutun nettijatkosta: [skrolli.fi/numerot](http://skrolli.fi/numerot)



Kasettilamerit

[kasettilamerit.fi](http://kasettilamerit.fi)

Kasettilamerit Facebookissa

[facebook.com/groups/kasettilamerit/](https://facebook.com/groups/kasettilamerit/)

Kasettilamerit Twitterissä

[twitter.com/kasettilamerit](https://twitter.com/kasettilamerit)

Break's Amiga Demo Scene Collection

[arabuisimiehet.com/break/amiga/](http://arabuisimiehet.com/break/amiga/)

Kestra Bitworld

[janeway.exotica.org.uk](http://janeway.exotica.org.uk)

The Commodore 64 Scene Database

[csdb.dk](http://csdb.dk)

TOSEC

[www.tosecdev.org](http://www.tosecdev.org)

Software Preservation Society

[softpres.org](http://softpres.org)

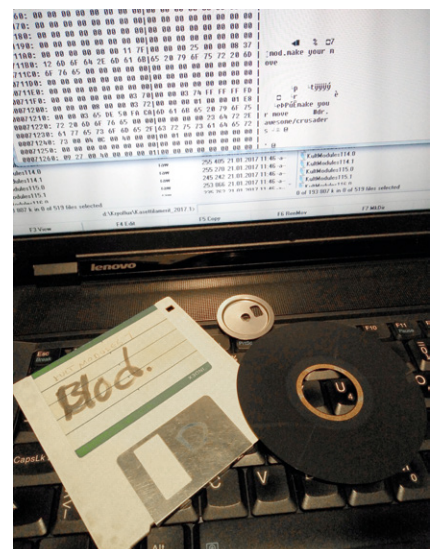
Demot, introt ja lehdet:

[warez.skrolli.fi/grue/](http://warez.skrolli.fi/grue/)

Kryoflux

[www.kryoflux.com](http://www.kryoflux.com)

tiedostoilla, ei ole enää meidän vallasamme. Näin on käynyt esimerkiksi netistä löytyvälle Amigan IPF-kokoelmalle, jonka harrastajat kasasivat saamistaan tiedostoista. 🐛



Tältäkin korpulta data saatiin pelastettua.

# Curta – mekaaninen taskulaskin

Teksti ja kuvat: Olli Wilkman

*Pippurimyllyn näköinen, kämmenelle mahtuva Curta on mekaanisten laskukoneiden eräänlainen huipentuma. Curta oli oman aikansa mullistava keksintö, kunnes transistorin ja mikrosirujen kehittyttyä elektroniset laskimet syrjäyttivät mekaaniset laskukoneet.*

**T**örmäsin itse sattumalta Curtaan työpaikallani Maanmittauslaitoksen Paikkatietokeskuksessa. Tiloissamme on useita lasivitriinejä, joissa esitellään vanhoja geodesian ja maanmittausalan kojeita. Vanhoja laskukoneita esittelevässä vitriinissä huomasin suurten pöytämallisten laskimien keskellä pienen mustan sylinterin. Matematiikan ja tietojenkäsittelyn historian ystävänä olin kuullut Curtasta jo aiemmin, ja sain sen ystävällisesti lainaan tutustuakseni siihen paremmin.

Kyseistä Curtaa lienee käytetty ensisellä Geodeettisella laitoksella kenttätöissä, mutta tarkempia tietoja sen käytöstä en ole löytänyt. Sarjanumeron perusteella se on valmistettu marraskuussa 1954.

## Natsi-Saksan varjossa

Curtan isä oli itävaltalainen insinööri **Curt Herzstark** (1902–1988). Hänen isänsä omisti laskukoneita maahantuvan ja valmistavan yrityksen Rechenmaschinenwerk AUSTRIA Herzstark & co, jossa myös Curt työskenteli koneiden korjaajana ja myyjänä. Isän kuoltua vuonna 1937 Herzstark johti yritystä.

Jo 1920-luvun lopulla Herzstark suunnitteli pientä, mukana kannettavaa laskinta. Tarve taskulaskimelle oli ilmeinen, sillä monella alalla tarvittiin tien päällä tai kenttäolosuhteissa laskuja, joihin laskutikun tarkkuus ei riittänyt. 1930-luvun lopulla hän keksi Curtan keskeisimmän osan: pyörivän rum-



mun, joka suoritti sekä yhteen- että vähennyslaskuja yhdeksän komplementti-aritmetiikan avulla. Nerokas rumpu korvasi yhdellä mekanismilla isojen laskukoneiden monimutkaisemmat koneistot. Herzstarkille myönnettiin taskulaskimen mekanismin keskeiset patentit vuonna 1939.

Toisen maailmansodan lähestyessä Herzstark ei kuitenkaan ehtinyt kehittää ideoitaan. Natsien liitettävä Itävallan osaksi Saksaa vuonna 1938 Herzstarkien yritys alkoi valmistaa armeijalle erilaisia kojeita. Herzstarkin isä oli juutalainen, joten hänen asemansa Natsi-Saksassa oli tukala. Vuonna 1943 eräs yrityksen työntekijöistä jäi kiinni ulkomaalaisen radiolähetysten kuuntelusta ja teloitettiin. Herzstark tuomittiin avunannosta ja lähetettiin Buchenwaldin keskitysleirille Saksaan.

Herzstarkin mekaanisen osaamisen vuoksi hän ei joutunut keskitysleirillä raskaimpaan pakkotyöhön vaan työskenteli konepajalla, joka valmisti mekaanisia kojeita sotavoimille. Myöhemmin osaston johtaja kuuli hänen suunnitelmistaan pienestä taskulaskimesta. Hänen käskettiin jatkaa sen

suunnittelua vapaa-ajallaan, sillä se olisi arvokas väline Saksan armeijalle. Herzstarkille vihjailtiin, että mikäli hän tuottaisi toimivan taskulaskimen, hänet ehkä palkkioksi määriteltäisiin uudelleen arjalaiseksi, mikä tekisi hänestä taas vapaan kansalaisen.

Herzstark suunnitteli Curtaa keskitysleirillä, kunnes sodan lopulla amerikkalaiset sotilaat vapauttivat leirin vuonna 1945. Laskimen piirustukset olivat käytännössä valmiit, ja Herzstark valmistutti kolme prototyyppiä saksalaisessa konepajassa. Neuvostoliiton armeijan lähestyessä hän matkusti prototyyppineen takaisin Itävaltaan etsimään rahoittajaa uuden yrityksen perustamiseksi. Liechtensteinin prinssi **Frans Josef II** halusi kehittää maansa teollisuutta 1940-luvun lopulla ja suostui rahoittamaan Herzstarkin keksintöä. Uusi yritys, Contina AG, perustettiin Liechtensteiniin, ja Curta-laskinten valmistus alkoi vuonna 1948. Herzstark jätti yrityksen vuonna 1952 ja Liechtensteinin suuri teollisuusyritys Hilti osti sen vuonna 1966, mutta Curtan valmistus jatkui aina vuoteen 1972 saakka. Laskimia valmistettiin yhteensä noin 140 000 kappaletta.



## Curtan käyttöliittymä ja perustoinnot

Curta-laskimen käyttö perustuu kolmeen numerorekisteriin: lukurekisteri koostuu kahdeksasta liukukytkimestä, joilla käyttäjä voi syöttää kahdeksannumeroinen luvun. Kun laskimen kampea käännetään yksi kierros, lukurekisterin sisältö lisätään 11-numeroiseen tuloslaskuriin. Samalla kuusi-numeroinen kierroslaskuri kasvaa yhdellä. Kierroslaskuri siis kertoo, kuinka monta kertaa lukurekisteri on lisätty tuloslaskuriin. Nostamalla Curtan kampi yläasentoon sen kääntäminen vähentää lukurekisterin arvon tulorekisteristä. Samalla myös kierroslaskurista vähennetään yksi.

Nollausrenkas on Curtan yläosassa oleva sokan näköinen rengas, jonka pyöryttäminen ympäri nolaa kierroslaskurin ja tuloslaskurin. Tarvittaessa vain jommankumman voi nolata; esimerkiksi kun halutaan laskea yhteen usean eri kertolaskun lopputulokset, kierroslaskurin voi nolata kertolaskujen välissä.

Tietyissä algoritmeissa, kuten jakolaskua suoritettaessa, on hyödyllistä laskea yhteenlaskujen sijasta vähennyslaskujen määrää. Tämän vuoksi Curtan kyljessä on kääntökytkin, joka muuttaa kierroslaskurin toiminnan käänteiseksi: yhteenlasku pienentää laskurin arvoa, ja vähennyslasku kasvattaa sitä.



Jotta kampea ei tarvitse veivata tuhansia kertoja, Curtan yläosassa on käännettävä positiovalitsin, jolla voi asettaa laskutoimituksen kertaluokan. Positiovalitsimen ollessa asennossa 1 tapahtuu kuten yllä: lukurekisteri lisätään tuloslaskuriin, ja kierroslaskuri kasvaa yhdellä. Kun positiovalitsin on asennossa 2, tuloslaskuriin lisätään lukurekisteri kymmenen kertaa, ja kierroslaskuri kasvaa kymmenellä. Positiovalitsimessa on yhteensä kuusi asentoa, mutta se on numeroitu yhteentoista asti. Yleisesti, kun positiovalitsin on asennossa k, kerroin on 10k. Siten jokaisessa positiovalitsimen asennossa jokaisen lukurekisterin liukukytkimien yllä oleva numero kertoo suoraan, monenteenko tuloslaskurin desimaaliin kyseessä oleva lukurekisterin numero lisätään. Kampea käännettäessä Curta suorittaa siis seuraavan laskutoimituksen: tulos = lukurekisteri × 10<sup>positiovalitsin-1</sup> + tulos.

Lisäksi lukurekisterin ja tuloslaskurin ohessa on joukko pieniä metallisia nuppeja, joita voi liu'uttaa edestakaisin kiskolla. Ne eivät vaikuta laskimen toimintaan, mutta niitä voi käyttää siirrettävinä desimaalierottimina tai merkitsemään muuta tärkeää numeroa.

## Yhdeksän komplementti

Curtan vähennyslasku perustuu yhdeksän komplementin aritmetiikkaan, joka mahdollistaa vähennyslaskut mekanismilla, joka suorittaa vain yhteenlaskuja. Yksi Curtan pienen koon mahdollistavista innovaatioista on, että yhdeksän komplementin laskeva mekanismi yhdistettynä yhteenlaskuun on kompaktimpi kuin erillinen vähennyslaskumekanismi.

Annetun luvun yhdeksän komplementti on luku, joka saadaan vaihtamalla jokainen luvun numero siihen numeroon, joka saadaan vähentämäl-



lä kyseinen numero yhdeksästä. Nollasta tulee yhdeksän, viitosesta nollan, ja niin edelleen.

Esimerkiksi: luvun 864 yhdeksän komplementti  $\overline{864} = 135$ .

Vähennyslasku A–B voidaan muuttaa yhteenlaskuksi laskemalla yhteen A:n komplementti ja B sekä ottamalla tämän summan komplementti (todistus jätetään harjoitustehtäväksi lukijalle).

Esimerkiksi:  $957 - 256 = \overline{(957 + 256)} = (042 + 256) = 298 = 701$ .

## Kertolasku

Kierroslaskurin lukema kertoo siis, kuinka monta kertaa lukurekisteri on lisätty tuloslaskuriin. Kertolaskun laskeminen tapahtuu siten, että asetetaan toinen (yleensä suurempi) tulon tekijöistä lukurekisteriin ja sitten lasketaan yhteen ja vähennetään sen kymmenmonikertoja kammen ja positiovalitsimen avulla, kunnes tulon toinen tekijä näkyy kierroslaskurissa. Jos tämä tuntuu monimutkaiselta, lue seuraava esimerkki.

Lasketaan kertolasku  $1337 \times 957$ . Asetetaan ensin liukukytkimillä lukurekisterin arvoksi suurempi luvusta, 1337. Sen jälkeen asetetaan positiovalitsin asentoon 4 (tuhannet) ja käännetään kampea kerran. Kierroslaskurissa lukee nyt 1000 ja tulorekisterissä 1337000.

Nostetaan kampi vähennyslaskumoodiin, käännetään positiovalitsin asentoon 3 (sadat) ja käännetään kampea kerran. Nyt tuloslaskurissa lukee 1203300 ja kierroslaskurissa 900.



Lasketaan kampi yhteenlaskumoodiin, positiovalitsin asentoon 2 (kymmenet) ja käännetään kampea viisi kertaa. Tuloslaskurissa on 1270150 ja kierroslaskurissa 950.

Käännetään positiovalitsin asentoon 1 ja käännetään kampea seitsemän kertaa. Nyt kierroslaskurissa lukee tulon toinen tekijä 957, ja tuloslaskurissa haluttu kertolaskun tulos, 1279509.

## Nyckpäivän harrastajat

Curta-laskin on saavuttanut tiettyä kulttimainetta. Netistä löytyy useita sivustoja, joilta löytää runsaasti lisätietoja, kuten mekanismin piirustuksia, huolto-ohjeita, algoritmeja eri laskutoimituksille ynnä muuta.

The Curta Calculator Page ([www.vcalc.net/cu.htm](http://www.vcalc.net/cu.htm)) on laaja kokoelma tekstejä, tiedostoja ja linkkejä. Sivuston kautta löytyvät muun muassa laskimen alkuperäiset tekniset piirustukset ja käyttöohjeet.

Hienomekaniikan harrastajalle Curtan jäljennöksen valmistaminen olisi erittäin haastava mutta ehkä mahdollisen rajoissa oleva projekti. ReCurta-projekti ([github.com/daeken/ReCurta](https://github.com/daeken/ReCurta)) ei ole päivittynyt pariin vuoteen, mutta sen tavoitteena oli tehdä modernit CAD-mallit kaikista Curtan osista alkuperäisten piirustusten perusteella. Tätä artikkelia varten tutkiessani en törmännyt yhteenkään valmiiseen jäljennökseen.

Marcus Wu ([wudev.digitaltorque.com](http://wudev.digitaltorque.com)) on mallintanut Curtan osat 4:1-mittakaavassa 3D-tulostimella valmistettavaan muotoon. Suurempaa mittakaavaa tarvitaan, koska muovisen 3D-tulosteen tarkkuus ei riittäisi mekanismin toimintaan. Muovisen version kestävyys on myös käytössä koetuksella. Sivulla on useita blogitekstejä mallin suunnittelusta. Wu on myös julkaissut 1:3-mittakaavaisen version mallista Thingiverse-sivustolla: [www.thingiverse.com/thing:1943171](http://www.thingiverse.com/thing:1943171). 🐞

*Katso video kertolaskusta Curtalla: [skrolli.fi/numerot](http://skrolli.fi/numerot)*





# Helppo langaton hiiri Amigalle

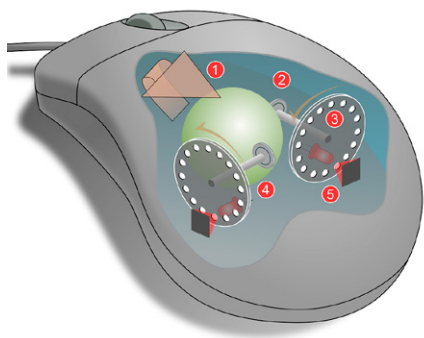
*Haluatko käyttää klassikkokonetta sohvalta ilman pitkiä jatkopiuhoja? Tämän ohjeen avulla voit muuttaa vanhat ohjaimet kauko-ohjaimiksi ilman yhtään juotosta tai koodiriviä.*

Teksti: Teemu Leppänen

Kuvat: Teemu Leppänen, Jeremykemp & Pbroks13/Wikimedia Commons

Ajatus langattomasta ohjaimesta Amigalle sai alkunsa, kun kesälomaprojektina kunnostin yli vuosikymmenen varastossa koskemattomana lojuneen Amiga 500:n. Tiedossa oli vain, ettei rakas ystävätär enää inahdakaan. Vaiherikas kunnostusprojekti olisi ihan oma juttunsa, mutta todettakoon, että apuun tarvittiin kavereita, netin Amiga-sivustoja, oskilloskooppia, yleismittaria, juotoskolvia ja epoksiliimaa. Päätin viimeistellä projektin muuttamalla koneen ohjaimet langattomiksi, jotta etenkin pelejä olisi mukavampi fiilistellä kotisohvalta.

Jutussa esitettävä langattoman ohjaimen idea toimii pienin muutoksinkin kaikissa Atari-tyyppisissä hiirissä ja digitaalisissa joystickeissa retroraudasta riippumatta. Atari-tyyppisiin joystickeihin on lehdessä tutustuttu aiemminkin (muun muassa Skrolli 2014.2 ja 2016.4), joten tässä jutussa keskitytään siihen, miten Amigan hiiri toimii. Joystick on joukko digitaalisia (1/0) kytkimiä, kun taas hiiren pallon liike tuottaa digitaalista signaalia vaihtelevalla taajuudella.

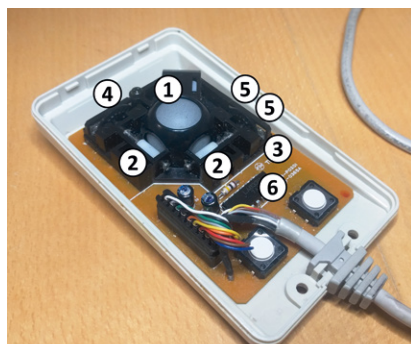


Kuva 1a. Mekaanisen hiiren läpileikkaus

## Hiiren toimintaperiaate

Tietokonehiiren konseptin kehittäjä **Douglas Engelbart** esitteli vuonna 1965 planimetriin perustuvan ratkaisun, jossa kahden toisiinsa kohtisuoran akselin liike siirrettiin potentiometreillä tietokoneelle ja näytöllä kursori liikkui vastaavasti. Ensimmäisenä pallon liikkeeseen perustuvan ohjaimen kehitti Telefunken vuonna 1968, mutta se ei katsonut tarpeelliseksi patentoida ideaa. Vuonna 1972 **Bill English** Xerox PARC-yhtiössä kehitti nykytekniikan mukaisen mekaanisen pallohiiren. Nimitys hiiri tulee ”hännästä” eli johdosta, joka liittyy sen tietokoneeseen.

Kuvassa 1a on mekaanisen hiiren läpileikkaus ja kuvassa 1b vastaavat komponentit Amigan perinteisessä ”tank mouse” -hiiressä. Hiiren liikuttelu pöydällä pyörittää sisällä olevaa palloa (1), joka vastaavasti pyörittää kahta toisiinsa nähden kohtisuorassa olevaa akselia (2), joilla saadaan määriteltyä x- ja y-akselit hiiren liikkeelle. Molempien akselien päässä on pyörivät rei’itetty levy, ja kumpaankin levyyn nähden kohtisuoraan asetettuna kaksi lediä (5, kuvassa näkyy yksi). Ledit lähettävät infrapunasignaalia kohti levyä, ja niiden jännite



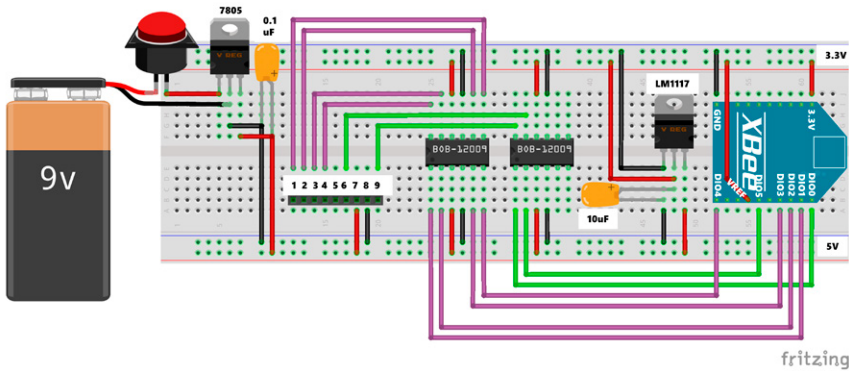
Kuva 1b. Amigan ”tank mouse” avattuna.

muuttuu sen mukaan, osuuko signaali levyyn vai reikään (4). Kun levyssä on pariton määrä reikiä ja ledit tarkasti toisella puolen levyä (180 asteen päässä toisistaan), ei ole mahdollista, että molemmat ledit generoisivat pulssiin saman jännitemuutoksen samaan aikaan. Näin ohjaimen liikkeen suunta saadaan selville siitä, kummasta ledistä tulevan pulssin jännitetaso muuttuu ensin (3). Hiiressä on vielä komparaattori (6), joka muokkaa ledien signaalit digitaalseksi ja vastaamaan tietokoneen ohjainportin jännitetasoja. Liikkeen nopeus saadaan selville pulssien tiheydestä eli jännitetaso muutoksien määrästä. Atari-tyyppisten hiirten liikettä tulisi näytteistää vähintään 300 hertsin taajuudella. Liikesignaalit tulkitsee Amigan Denise-piiri, ja hiiren painonappeihin taas reagoivat CIA ja Paula.

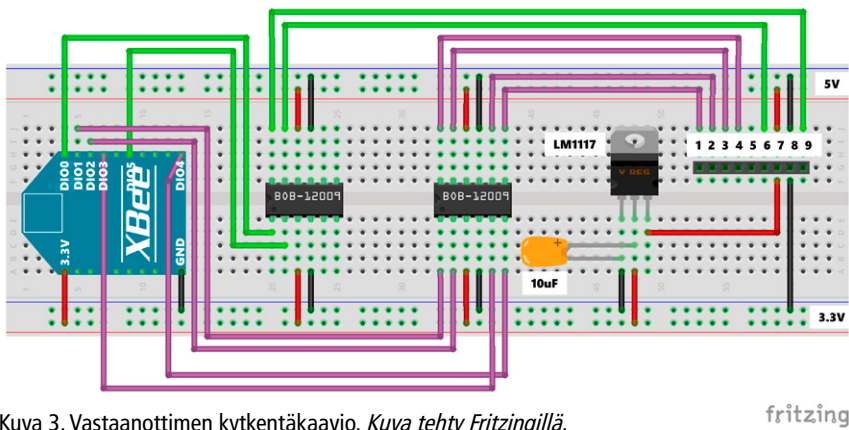
## Simppeleäkö?

Idean juju on, ettei langattoman ohjaimen toteuttamiseksi tarvitse kirjoittaa riviäkään koodia tai edes juottaa (rautaprojektille epätyypillisesti). No mitäs hauskaa tässä sitten on? Tavoitteena onkin esittää yksinkertainen ja yleinen langaton ratkaisu kaikille Atari-tyyppisille ohjaimille. Aiemmissä ratkaisuissa tiedonsiirtoon on käytetty infrapuna- tai Bluetoothia, jotka vaativat mikrokontrollerin tulkkamaan signaalit ohjaimen liikkeiksi ja ohjaamaan ohjainportin pinnejä. Niiden toteutus on monimutkaisempi, mutta etuna on, että analogisista ohjaimista pystytään tekemään langattomia ohjelmallisesti AD-muuntimien avulla.

Olen aiemmin tutustunut langattomiin sensoriverkkoihin esineverkkojen maailmassa, ja sieltä on tullut tutuksi XBee-tiedonsiirtoteknologia ([www.digi.com/products/xbee-rf-solutions](http://www.digi.com/products/xbee-rf-solutions)). XBee-radioihin on integroitu I/O-pinnejä, joihin lähetinpäässä kytketään ohjaimelta tulevat signaalit ja vastaanottopäässä ohjainporttiin menevät signaalit. XBeen line passing-ominaisuuden avulla lähetinpään tulonastojen digitaalinen tila (1/0) kopioituu lähes reaaliajassa vastaanotinpään output-pinnien tilaksi. Näin, ilman riviäkään koodia, radio hoitaa hiiren liikkeen välityksen langattomasti ohjaimelta porttiin. XBee-radioita on aiemmin käytetty muun muassa DIY-joystickien kanssa ja RC-laitteiden ohjaamiseen.



Kuva 2. Lähettimen kytkentäkaavio. Kuva tehty Fritzingillä.



Kuva 3. Vastaanottimen kytkentäkaavio. Kuva tehty Fritzingillä.

Ohjaimen pinnit	Logiikkatason muunnin BOB-12009	XBee-radio	Logiikkatason muunnin BOB-12009	Amigan portin nastat
1	HV4 LV4	DIO4	LV4 HV4	1
2	HV3 LV3	DIO3	LV3 HV3	2
3	HV2 LV2	DIO2	LV2 HV2	3
4	HV1 LV1	DIO1	LV1 HV1	4
5	-	-	-	5
6	HV4 LV4	DIO5	LV4 HV4	6
7	-	-	-	7
8	-	-	-	8
9	HV3 LV3	DIO0	LV3 HV3	9

Taulukko 1. Ohjaimen ja portin pinnien kytkennät, kun käytössä on kaksi nelikanavaista logiikkatason muunninta.

Komponentttilistaus	
2 kpl	XBee S2C -radio
2 kpl	XBee-liitinadapteri radion 2 mm:n nastavälin sovittamiseksi 2,54 mm:n nastaväliin
4 kpl	Kaksisuuntainen nelikanavainen logiikkatason muunnin 5 V 3,3 V: BOB-12009 tai vastaava
1 kpl	5 voltin regulaattori: 7805 tai vastaava
2 kpl	3,3 voltin regulaattori: LM1117 tai vastaava
1 kpl	D-liitin, 9 napaa, uros
1 kpl	D-liitin, 9 napaa, naaras
1 kpl	0,1 µF polarisoitu kondensaattori
2 kpl	10 µF polarisoitu kondensaattori
1 kpl	9 voltin paristoliitin tai vastaava
2 kpl	Reikälevy tai koekytkentälevy

Taulukko 2. Lähettimen ja vastaanottimen vaatimat komponentit.

## Kytchentäkaavio

Ohjaimen ja ohjainportin liittämiseksi lähetin- ja vastaanotinradioihin täytyy kasata kaksi yksinkertaista piirilevyä.

Kuvassa 2 on lähettimen kytkentäkaavio. Oletuksena virtalähteenä on 9 voltin ladattava paristo, jolloin tarvitaan kaksi jänniteregulaattoria: 5 V ohjaimen kytkentään ja 3,3 V XBee-radiolle. Kuvan koekytkentälevyssä yläosaan viedään jännite 3,3 V ja alapuolelle 5 V. Regulaattorien kondensaattorien arvot on saatu suoraan datalehdiltä. Kun lähetin tehdään joystickille, on mahdollista käyttää pelkästään 3,3 V:n jännitettä ja jättää 5 V:n regulaattori sekä logiikkatason muunnin pois. Pariston säästämiseksi tarvitaan virtakytkin.

Piirilevylle ohjaimen kytkee 9-napainen D-liitin. Eri retro-ohjaimissa liittimen pinnijärjestys hieman vaihtelee, esimerkkinä Atari ST:n hiiri, mutta niissä tarvitsee vain kytkeä vastaavat linjat. Yleisesti ohjaimen signaalien välittämiseen tarvitaan kuusi I/O-pinniä: kaksi painonappia sekä suunnat eteen, taakse, vasemmalle ja oikealle. Kuvassa D-liittimen linjat on kytketty Amigan ohjainmäärittelyn mukaan. Painonappien linjat (6 ja 9) on kuvattu vihreällä ja suuntien linjat (1–4) vaaleanpunaisella. XBeen I/O-nastat DIO0–DIO5 on valittu numerojärjestyksessä, mutta muikin kytkentä olisi mahdollinen. Nastoilta 7 (+5 V, punainen) ja 8 (0 V, musta) saadaan ohjaimelle jännite. Lisäksi signaalien jännitetaso pitää muuttaa XBee-radiolle sopivaksi: 5 voltista 3,3 volttiin.

Kuuteen linjaan tarvitaan joko yksi kahdeksankanavainen tai kaksi nelikanavaista logiikkatason muunninta. Kytkentäkaaviossa on käytetty transistorista rakennettua komponenttia BOB-12009 (valmistaja Sparkfun). Muitakin voi käyttää – esimerkiksi 74-sarjassa on useita muuntimiksi sopivia piirejä, mutta ne tarvitsevat lisää ohjauslinjoja ja kuluttavat enemmän virtaa. Nastojen kytkennät on esitetty tarkemmin taulukossa 1 ja komponenttiluettelo taulukossa 2.

Kuvassa 3 on vastaanottimen kytkentäkaavio. Vastaanottimen virrat saadaan Amigan ohjainportin (CN1 tai CN2) kautta. Rajoituksena on, että portti antaa virtaa maksimissaan 50 mA, josta XBee S2C tarvitsee da-



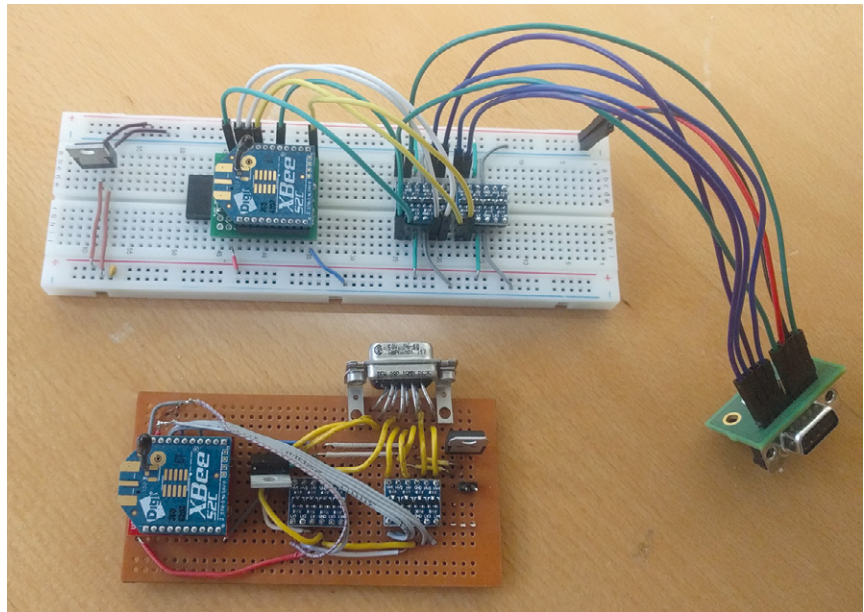
talehden mukaan vastaanottotilassa enintään 31 mA. Radion I/O-nastat (DIO0–DIO5) on kytketty logiikkatason muuntimien BOB-12009 kautta Amigan ohjainportin vastaaviin nastoihin (1–4, 6 ja 9). Virrankulutus rajaa 74-sarjan piirit pois, sillä niitä testatessa vastaanottimen kokonaisvirrankulutus oli luokkaa 55 mA.

Kuvissa 4a–4c on toteutettuja prototyypppejä. Lähetin vie virtaa ”tank mousen” kanssa noin 70 mA, uuden hiiren kanssa noin 50 mA ja joystickiin integroituna alle 40 mA. Langatonta ohjainta olisi siis mahdollista käyttää tunteja ilman pariston vaihtoa. Tee-se-itse-joystickin lähettimen voisi toteuttaa kolikkoparistolla (3 V), mutta sellaisten varaus (noin 200 mAh) ei riitä kovin pitkään. Vastaanotin vie virtaa alle 35 mA.

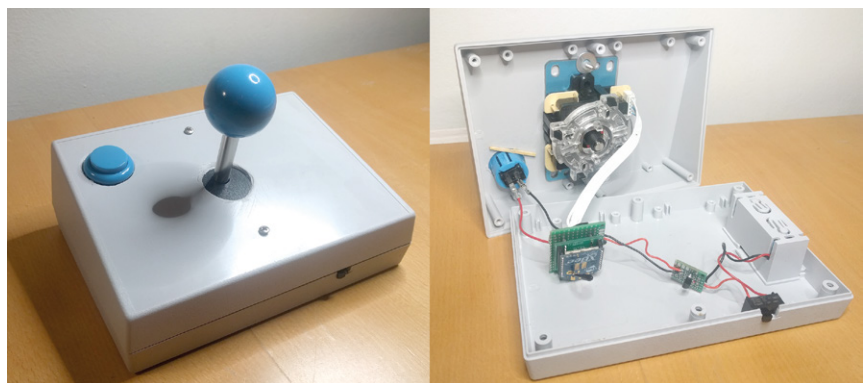
### Radion määrittäminen

Radioiden ja lisäksi tarvittavan USB-ohjelmointilaitteen hankinta onnistuu helposti verkkokaupoista. Tässä on tärkeää huomata, että vaadittu line passing -ominaisuus toimii ainoastaan IEEE 802.15.4 -protokollapinnalla, jota kaikki myynnissä olevat XBee-mallit eivät tue. Itselläni on käytössä S2C-malli, jota myytiin esiohjelmituna toiselle protokollalle. Radion laiteohjelmisto piti päivittää versioon 802.15.4 TH.

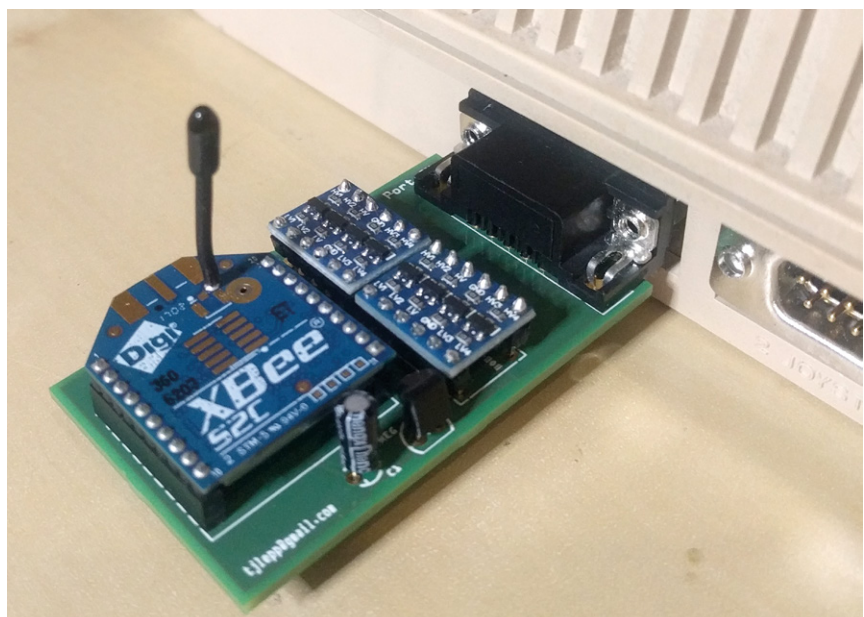
Valmistaja tarjoaa ilmaisen ja monipuolisen XCTU-ohjelman (kuva 5), jolla voi määrittää radioiden asetukset, seurata verkkoliikennettä ja myös päivittää laiteohjelmiston. Jotta line passing -ominaisuus saadaan käyttöön, tulee lähetin- ja vastaanotinradiot määrittää taulukon 3 mukaisesti. Baudinopeuden kannattaa olla maksimi eli 230 400, koska viestien välitys tapahtuu sarjaliikenteenä. Radion osoite löytyy painettuna sen pohjasta: esimerkiksi oman lähetinradion 64-bittinen osoite on 0013A2004106FF78. Konfiguroinnin jälkeen lähetin huomaa ohjaimen kytkettyjen kuuden tulonastan (DIO0–DIO5) tiloissa tapahtuvat muutokset ja lähettää automaattisesti uudet tilat vastaanottimelle, joka taas asettaa vastaavat lähtönastat samaan tilaan. XBee-radioissa olisi myös virtaa säästäviä sleep-tiloja, mutta niitä käyttäen tulonastojen näytteistys olisi hitaampaa.



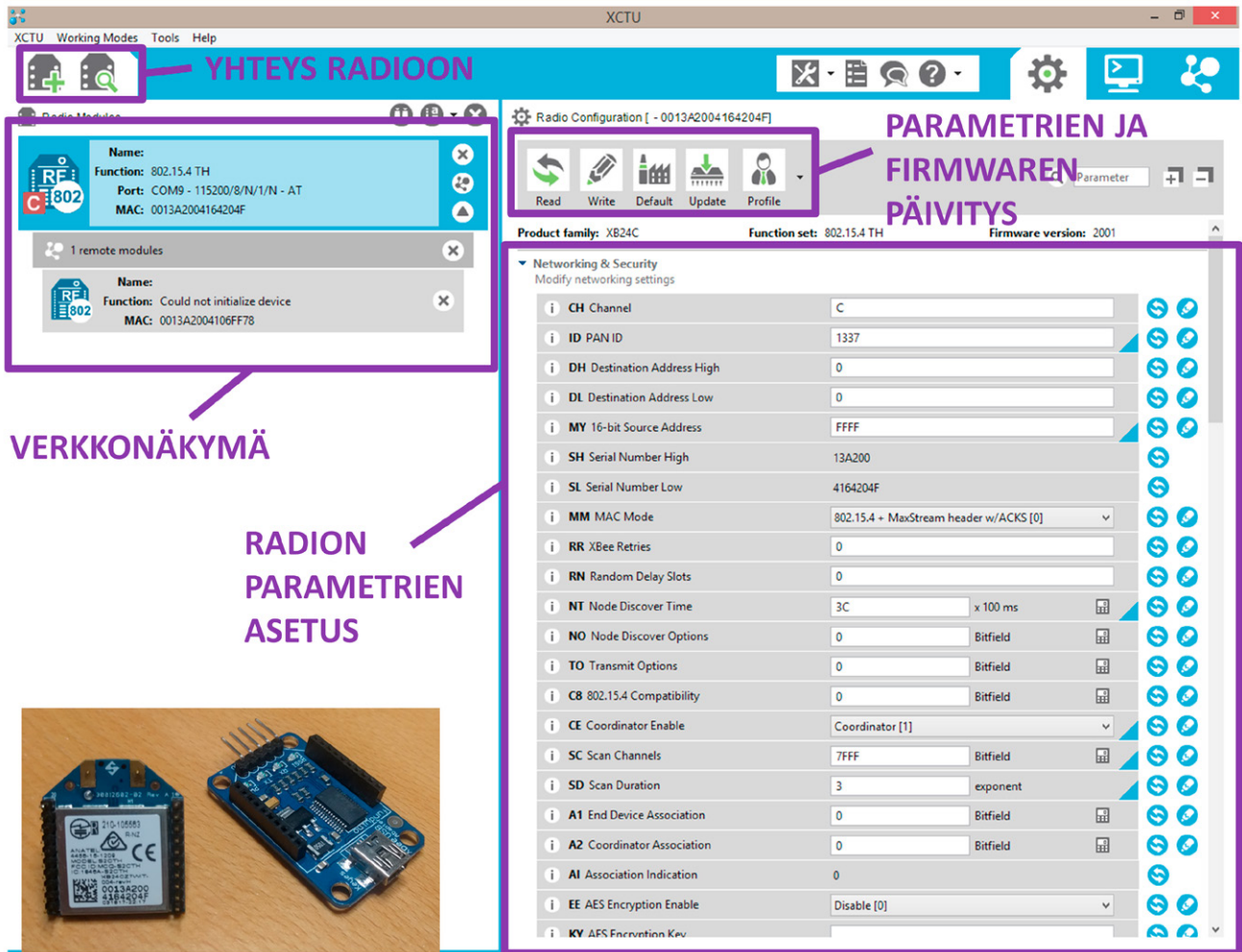
Kuva 4a. Yllä vastaanotin koekytkentälevyllä ja alla lähetin piirilevyllä, mallia itse tehty ja huolella korjailtu.



Kuva 4b. Jämaosista koottu langaton DIY-joystick. Koneisto ja 9 voltin paristo vaativat ison kotelon, mutta malli on hyvin sylissä pidettävä.



Kuva 4c. Valmistetulle piirilevyllä toteutettu vastaanotin liitettyä Amiga 500:n ohjainporttiin.



Kuva 5. XCTU-ohjelman käyttöliittymä. Alakulmassa XBee-radiion pohja, jossa osoite näkyvissä, sekä USB-ohjelmointilaite. Lue lisää osoitteesta [learn.sparkfun.com/tutorials/exploring-xbees-and-xctu](http://learn.sparkfun.com/tutorials/exploring-xbees-and-xctu)

PARAMETRI	LÄHETIN	VASTAANOTIN	SELITYS
BD	230400	230400	Baudinopeus
CH	C	C	Käytetty kanava
ID	1337	1337	Oman likiverkon tunnus
DH	0013A200	-	Vastaanottimen osoitteen ylimmät 4 tavua, aina vakio 0013A200
DL	nnnnnnnn	-	Vastaanottimen osoitteen alimmat 4 tavua
MY	FFFF	FFFF	Vakioasetus FFFF
DIO5	DI [3]	DO High [5]	Lähettimen I/O-nasta 5 tuloksi (DI). Vastaanottimen I/O-nasta 5 lähdeksi (DO)
DIO4	DI [3]	DO High [5]	Lähettimen I/O-nasta 4 tuloksi (DI). Vastaanottimen I/O-nasta 4 lähdeksi (DO)
DIO3	DI [3]	DO High [5]	Lähettimen I/O-pinni 3 tuloksi (DI). Vastaanottimen I/O-pinni 3 lähdeksi (DO)
DIO2	DI [3]	DO High [5]	Lähettimen I/O-pinni 2 tuloksi (DI). Vastaanottimen I/O-pinni 2 lähdeksi (DO)
DIO1	DI [3]	DO High [5]	Lähettimen I/O-pinni 1 tuloksi (DI). Vastaanottimen I/O-pinni 1 lähdeksi (DO)
DIO0	DI [3]	DO High [5]	Lähettimen I/O-pinni 0 tuloksi (DI). Vastaanottimen I/O-pinni 0 lähdeksi (DO)
IC	3F	-	Lähettimen pinnit tilaan "tunnista tilamuutos"
IA	-	0013A200nnnnnnnn	Lähettimen koko 64-bittinen osoite

Taulukko 3. XBee-radioiden tiedonsiirtoparametrit line passing -toiminnolle.

## Testidataa

Insinööriä tietysti kiinnostaa ratkaisunsa toimivuus. Tarkoituksena oli selvittää, kuinka hyvin radio pystyy näytteistämään nopeaa ohjaimen signaalia (yli 300 Hz), ja miten hyvin vastaanotin pää sitä seuraa. XBeen manuaali ei yksiselitteisesti kerro maksiminäytteenottajaajuutta line passing -ominaisuudelle.

Hiiren liikkeitä testattiin sekä hitaila että nopeilla kädenliikkeillä, jotka siis tuottavat digitaalista signaalia eri taajuuksilla. Testaus tehtiin kolmella eri ohjaimella: alkuperäisellä Amigan tankkihiirellä, uudella Amigan hiirellä (malli 327124) sekä alkuperäisellä TAC-2-joystickilla. Käsivaralla tehdyissä testeissä todettiin, että hiiri tuottaa akselien liikettä ainakin 1000 hertsin taajuuksilla. Hiiressä olevan komparaattoriin (vanhassa hiiressä LA6339 ja uudessa LM339) reaktioaika on datalehden mukaan noin 1 mikrosekunti.

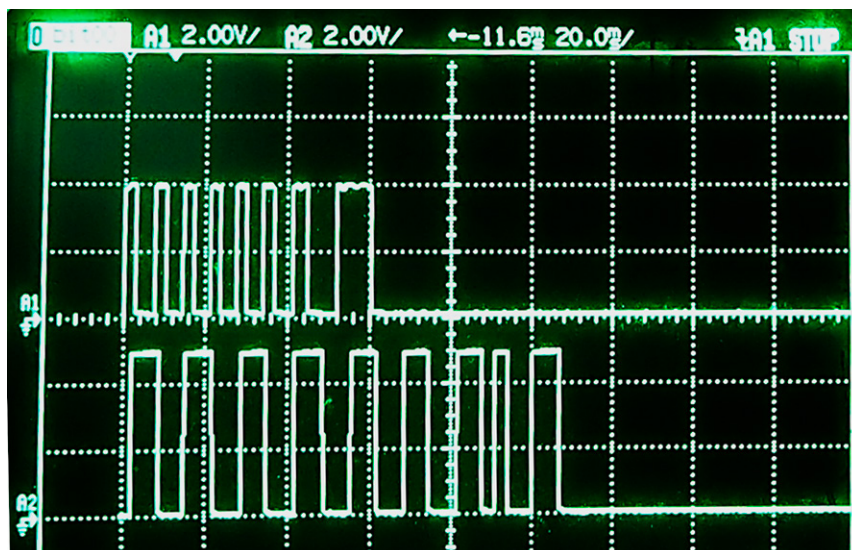


Testatessa vanhaa hiirtä jäi tuntu- ma, että kursori laahasi ruudulla aavistuksen jäljessä. Toisinaan kursori hetkellisesti heilui itseksen liikkeen suuntaisesti, mikä vaikuttaa käyttökokemukseen. Hitaisissa liikkeissä hiiri kuitenkin toimii käyttökelpoisuuden rajamailla. Nopeissa pitkissä liikkeissä kursori saattoi jämähtää paikalleen hetkeksi kesken liikkeen tai seurata selvästi perässä. Uudella hiirellä ei ratkaisu oikein toiminut. Hitailta liikkeillä kursori liikkui hapuilevasti valittuun suuntaan, mutta nopeilla liikkeillä se jäi tärise- mään paikalleen. Syy tähän lienee se, ettei radio pystynyt seuraamaan liike- signaalia riittävän tarkasti. Oskilloskoopista katsottuna näytteenottataajuus XBeellä oli luokkaa 150–200 Hz.

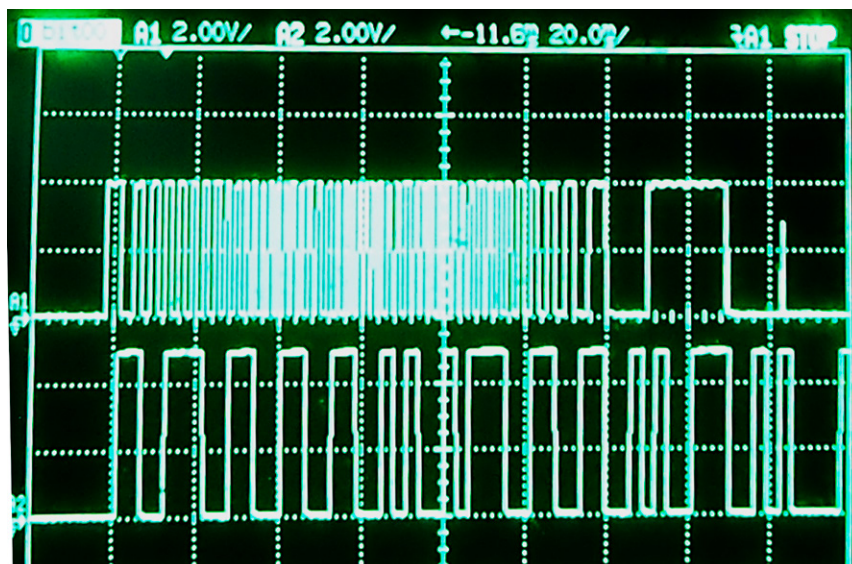
Langatonta TAC-2-joystickia tes- tattiin *Summer Games* -pelin kahdes- sa lajissa: pientä tarkkuutta vaativa skeet-ammunta sekä veivausta vaativa 100 metrin juoksu. Joystickilla tulok- siin voi olla tyytyväinen: pelatessa ei havaittu mitään poikkeavaa, ja vas- taanotin seurasi muutaman millise- kunnin viiveellä. Satasen juoksussa päästiinkin reilun 9 sekunnin juoksu- aikaan, kun aikoinaan treenattu lihas- muisti pikkuhiljaa heräsi. Oskilloskoopista seuraten 10–20 millisekunnin (50–100 Hz) näytestitys olisi riittävä pelaamiseen, mutta hifistit voivat olla asiasta eri mieltä.

Lopputuloksena voi todeta, etteivät näytestityksen ja langattoman tiedon- siirron viiveet kevyttä pelaamista häi- rinneet. Bonuksena valmistaja lupaa XBee S2C -radiolle näköyhteydellä yli kilometrin kantaman. Vaikka projekti ei nyt täydellinen menestys ollutkaan, tar- josi se mukavia nostalgisia hetkiä integ- roidessa nykyteknologiaa lapsuusmuis- toihin. Olen kiinnostunut vaihtamaan ajatuksia projektista – esimerkiksi siitä, miten hiiret saisi toimimaan paremmin, ja miten projektia voisi jatkokehittää. Minut saa kiinni sähköpostilla osoit- teesta [tjlepp@gmail.com](mailto:tjlepp@gmail.com). 📧

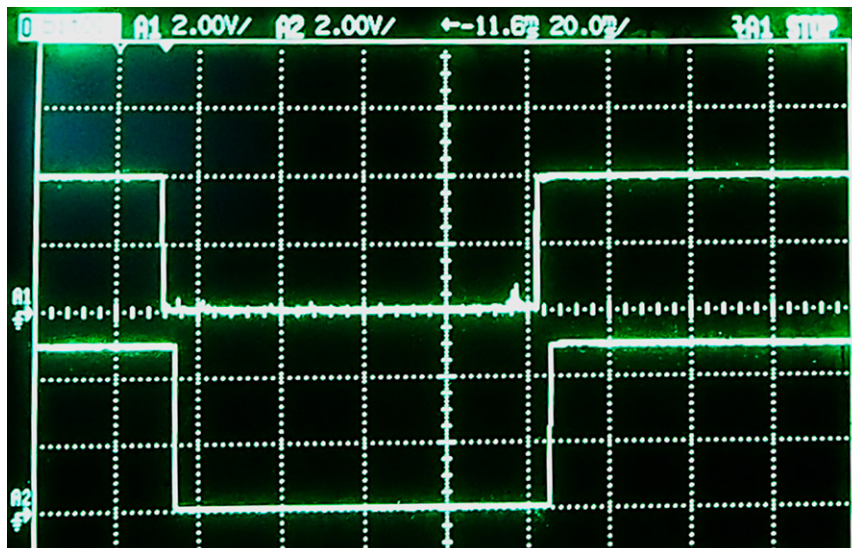
Kiitokset Pyksylle TAC-2-joystickin lainasta sekä FabLab Oulun henkilö- kunnalle avusta kytkentäkaavioiden tarkastamisessa.



Kuva 6a. Ruutukaappaus oskilloskoopista tankkihiiren normaalilla liikkeellä. Yllä lähetin (A1) ja alla vastaanotin (A2). Nähdään, että vastaanotinpäässä signaalia pystytään seuraamaan hyvin 20–40 ms:n viiveellä, joskaan ei aivan täsmällisesti.



Kuva 6b. Vastaava mittaus uuden hiiren nopealla ja pitkällä liikkeellä. Nähdään, että hiiri tuot- ta signaalia paljon korkeammalla taajuudella kuin XBee pystyi välittämään.



Kuva 6c. Mittauksessa testattiin TAC-2-joystickia. Nähdään, että vastaanotin seuraa signaalia varsin hyvin muutaman millisekunnin viiveellä.



1990

MIKÄ TÄMÄ ON?  
NÄYTTÄÄ VAIKEALTA.

EN OSTANUT. VIRHE!

1991

HEI, ONKO TÄMÄ SE PELI?

AMIGA FORUM

1992

TÄMÄ OLISI MAHTAVA! AIVAI..

CHAOS STRIKES BACK -LÄPIPELUOHJE

1992

NO NYT.. MUTTA SE ON PC:LLE, JA MAKSAISI MUUTENKIN LIIKAA.

SITTEEN ERÄÄNÄ ILTANA:

! TURBO RAKETTI  
ERÄIN MEGA DISK

HEI, SULLA ON TÄÄLLÄ DUNGEON MASTER!!!

VOIDAANKO KOPIOIDA SE MULLE?

VAIKKA. ITSE EN TUOSTA PELISTÄ NIIN VÄLITÄ.

HMM, X-COPY EI NYT OSAA.

HIBBLECOPY 00 0 79 1 BOTH  
START REPEAT STOP NAME DEFAULT  
SOURCE DRIVE UPPER SIDE  
0 000000077  
1 000000077  
2 000000077  
3 000000077  
4 000000077  
5 000000077  
6 000000077  
7 000000077  
8 000000077  
9 000000077  
TARGET DRIVES

MÄ HALUAN SITTEEN OSTAA SEN.

JAA...

50 MARKALLA MYYN.

OKEI.

VIIMEINKIN!

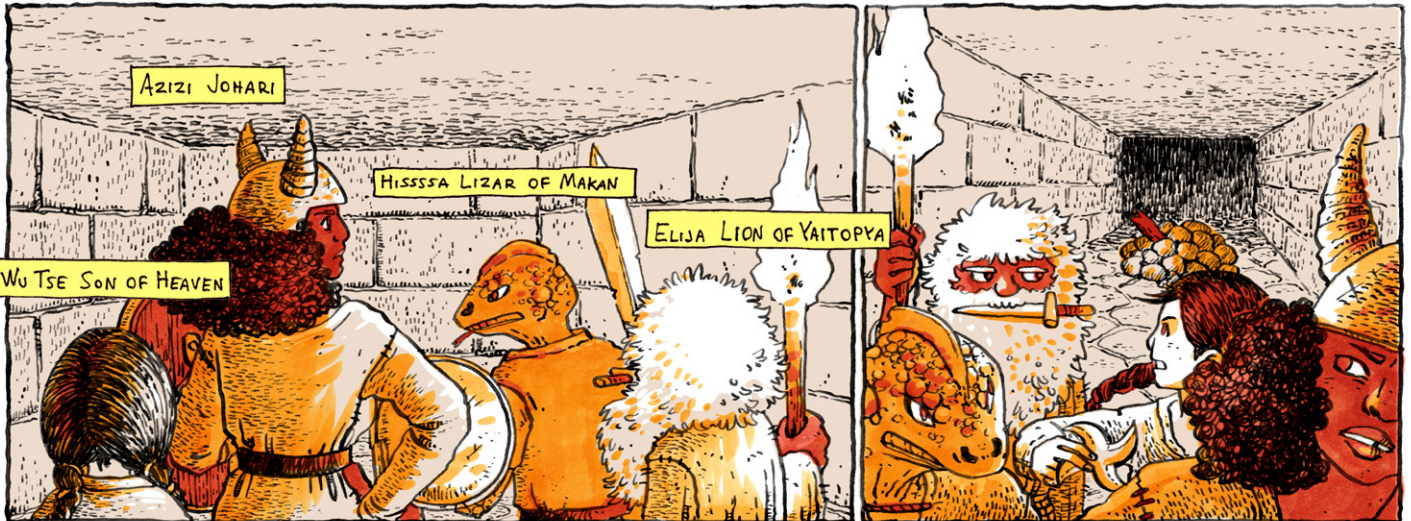
MUUMIOT, ALTAR OF REBIRTH, TYHJÄT PULLOT, OUDOT KÄÄRÖT, MITÄ SE KAIKKI TARKOITTI? ILMAN OHJEKIRJAA KAIKKI OLI SITÄKIN MYSTISEMPÄÄ.

RESURRECT VAI REINCARNATE? PELIT -LEHDESSÄ SANOTTIIN, ETTÄ REINCARNATE ON PAREMPI...

..MUTTA EKAN HAHMON JÄLKEEN PITÄÄ VALITA MUUT HAHMOT NOPEASTI..

MUUTAMAN HARJOITUS-ALUN JÄLKEEN SEIKKAILU VIIMEIN ALKOI!





"UMPIKUJA! KÄÄNNYTÄÄN ÄKKIÄ TAKAISIN!"

"MYÖHÄISTÄ, KIVIKASAT OVAT TUKKINEET KÄYTÄVÄN!"



"OLISIPA JOUSKARI!"



"NIITÄ ON LISÄÄ!"



LOPULTA LORD CHAOSKIN KUKISTUI, JA OLEN PELANNUT PELIN LÄPI 3-4 KERTAA.

MYÖS SISKONI INNOSTUI, JA OHJEISTIN HÄNTÄ PELIN LÄPÄISYSSÄ. SIENIÄ HAKATTIIN MONTA PÄIVÄÄ, ENNEN KUIN USKALSIMME NELOSKERROSTA SYVEMMÄLLE.

CHAOSIA EN OLE KOSKAAN LÄPÄISSYT. KUOLIN PUOLESSA VÄLISSÄ NÄLKÄÄN.

RAKASTAN TÄTÄ GENREÄ. BLACK CRYPT, BEHOLDERIT, KNIGHTMARE, GRIMROCKIT, HIE-NOJA KAIKKI.

TÄSSÄ ON HUVIKSENI TYÖSTÄMÄ MOCKUP. SE JÄI KESKEN.



# Huuda kovempaa

Mikä niissä kauhuelokuvissa ja -peleissä oikein viehättää? Skrollin yleensä kaikkietävä toimituskin joutuu antautumaan tämän kysymyksen edessä. Tyytykäämme siis kummastelemaan muutamia valkokankaalta pelimuotoon siirtynyttä kauhisteluja ja miettimään, millaisista aineksista hyvä kauhupeli oikein koostuu?

Teksti: Jukka O. Kauppinen

Kuvat: Jukka O. Kauppinen, Mobygames, Sega, Konami, GMH



## The Texas Chainsaw Massacre

VSS / Wizard Video Games, 1983  
Atari 2600

Jos Atari 2600 oli kaikkien pelikonsoalien (kaupallisesti onnistunut) esi-isä, niin sille julkaistu *Teksasin moottorisahamurhat* -elokuvaan pohjautuva peli oli samoin yksi varhaisimmista kauhupelieistä. Ehkä jotain vanhempaakin löytyy, mutta ei paljoa.

Tässä moottorisahasimulaatiossa tilanne käännettiin myös reteästi toisinpäin. Kauhuelokuvissa esiintyy usein hirviö tai murhaaja, jolta yritetään päästä pakoon. Nytpä pelaaja olikin itse reipas työmies, joka jahtaa **Leatherfacen** tontille eksyneitä turisteja. Näin pysyy tontti puhtaana muiden jätöksistä, joskin Nahkanaama itse on aikamoinen sottapytty. Tontilla kun voi satuttaa itsensä muun muassa pyörätuoleihin ja lehmien pääkalloihin.

Karultahan se näyttää ja tuntuu, mutta pisteet ajankuvaan nähden kovalle yritykselle ja rohkealle kuvakulmalle.



Vanhoilla pikseleillä ei veri vielä roiskunut.



Oliko kauhu siedettävämpää verikapselia pureskelemalla?

## Friday the 13th

Domark, 1986  
Amstrad CPC, Commodore 64, ZX Spectrum

Brittiläisen Domarkin tulkinta *Friday the 13th* -elokuvasta jäi pelihistoriaan kahdesta syystä. Ensinnäkin sen markkinointikampanja oli nerokas, aggressiivinen ja tunteita herättävä. Huikean kansigrafiikan, raivokkaan mainonnan, kauhuääniskaban ja onnistuneen ”hei osta peli, saat kaupan päälle verikapselita!” -promootion yhdistelmä myi peliä hurjasti samaan aikaan, kun kukkahatut lentelivät kassakoneiden aiheuttamassa tuulessa.

Siinä sivussa itse peli oli aika kamala. Kamala, kiireessä tehty ja silti yllättävä. Nuorisolauma oli hukassa Crystal Laken lomaleirillä, ja yksi porukasta onkin naamioitunut joukkomurhaaja **Jason**. Hän tosin näyttää samalta kuin

muutkin teinit kunnes ottaa macheten esiin ja aloittaa tappotyöt. Pelaajan täytyy tunnistaa ja tappaa Jason ajoissa, tai ainakin ennen kuin joutuu tämän yllättämäksi.

Kehnon pelattavuuden alla piilee kuitenkin enemmän kuin äkinäinen pelaaja huomaakaan. Pelissä on yllättävää dynaamisuutta ja outouttakin. Jason näet esimerkiksi pelkää ristiä, jos sellaisen keksii rakentaa, ja hänen jahtaamisessaan on monia jänniä detaljeja. Muistikuvieni mukaan pelaajakin voi muuttua Jasoniksi, mutta sitä en muista, oliko se tarkoituksellista vai bugi.

Ei *Fridaystä* laatua saa ulos puristamallakaan, mutta se on silti kiinnostava näkökulma kasibittiseen kauhuun ja yhteen ensimmäisistä rankasti hypetyistä videopelieistä. Sellaisena tsekkaamisen arvoinen.





Eräänlainen VR-kypärä tämäkin?



Pimeitä käytäviä. Liikkeen tunnistin. Täydellinen tappaja.



Aina voit yrittää paeta. Mutta kauasko pääset?

## Saw

Zombie Studios / Konami, 2009  
PC, PlayStation 3, Xbox 360

*Saw*-elokuvat ovat kamaa, jollaisesta en ole koskaan kyennyt nauttimaan. Ihmisten silpominen ja raaka väkivalta sen itsensä takia menevät minulta yli, mutta moni tykkää, ei siinä mitään. Videopelinä sen kauhut jäävät kuitenkin kovin etäisiksi.

Ehkä syy oli peliversion ajankohta. Moni peli näytti vajaa vuosikymmen sitten hyvältä, mutta pikkutarkka ihmisen kuvaaminen oli yhä vajaata. Niinpä rujojen ja tökeröiden polygonijanttereiden silpominen ei yltänyt lähellekään *Saw'n* ydintä: julmaa body horroria.

Kauhuton kauhupeli, joka ei pelota tai ällötä, ja jonka toimintakin on mäl-sää. Ei näin.



## Alien: Isolation

Creative Assembly / Sega, 2014  
Linux, Mac, PC, PlayStation 3, PlayStation 4, Xbox 360, Xbox One

Jos viimeisen vuosikymmenen ajalta halutaan etsiä yksi peli, joka ei yltänyt kaupallisesti sille kuuluvaan gloriaan, niin se on *Alien: Isolation*. Peli, joka ei ole pelkästään kauhua – mutta joka parhaimmillaan on sitä mitä hyytävimmällä tavalla. Brittistudion uskomattoman kaunis ja jäätävän tunnelmallinen science fiction -avaruuskauhu lainasi inspiraationsa ja teemansa ensimmäisestä *Alien*-elokuvasta, joka oli loppujen lopuksi nimenomaan kauhuleffa.

Kehnomman toiminnan ja mälsien puzzlejen välissä *Isolation* onnistuu tiivistämään myös *Alienin* pelottavuuden täydelliseksi interaktiiviseksi kokemukseksi. Ympärillä hajoavan avaruusaseman nitinat ja kolahdukset saattavat peittää alleen Pedon liikehdintää. Heilahtiko varjossa sen häntä? Uskomattoman pelottavat hiiviskelykohotukset Pedon ollessa lähellä nostavat hien pintaan, ja se tunne, kun yhtäkkiä tajuat *Alienin* istuvan juuri sen pöydän päällä, jonka alla kökötät hengittämättä piilossa...

*Isolationissa* meillä oli täydellinen *Alien*. Kesyttämätön, hirvittävä ja armoton. Yksi ainoa mutta sitäkin vaarallisempi. Parhaimmillaan paras kauhupeli ikinä.

## Friday the 13th: the Game

IllFonic LLC / Gun Media Holdings, 2017  
PC, PlayStation 4, Xbox One

Menihän siihen yli 30 vuotta, mutta *Friday the 13th* -elokuvan paluu peliruuduille – mobiiliroskaahan ei laske – onnistui poikkeuksellisella tavalla. Puhtaasti nettipeliksi tehty *13th* näet rakentaa hienosti seikkailun, joka pursuaa sosiaalista selviytymiskauhua ja Jasonin pelkoa.

Pelin asetelma on klassinen: lauma elämäniloisia nuoria lomailee Crystal Laken kesäleirillä, kunnes Jason saapuu ja tappaa. Alkaa yleensä 5–15 minuuttia kestävä hyytävä pyrähdys, kun leiriohjaajapelaajat yrittävät löytää pelastusvälineitä ja vältellä Jasonia. Jasonin pelaaja taas hyödyntää supervoimiaan, seuraa leiriohjaajia ja tekee näistä selvää yksi kerrallaan, jos vain pystyy. Mutta eivät jonnetkaan aivan avuttomia ole, ja pelastuminen on mahdollista, etenkin jos pelaajilla on onnea tai he tekevät hyvää yhteistyötä.

Vaikka peli on modernein silmin nähtynä kömpelö, se on kuitenkin sympaattisen kömpelö. Pakeneminen, piilottelu, turvapaikkojen rakentaminen ja kauhun elementit tekevät siitä samalla myös hyvin tunnelmallisen, jopa alkuperäisteosta ja ylipäätään klassista mökkikauhua kunnioittavan teoksen. Ei hullummin indiepeliltä, jonka kaikki odottivat epäonnistuvan totaalisesti. 🤘



# SE AINOA OIKEA PELIOHJAIN

## Kilpailun ammattilainen kertoo

Teksti: Ville Jouppi  
Kuvat: Manu Pärssinen,  
Ville Jouppi

*Kuten kaikki 80- ja 90-luvuilla kotimikroilla pelanneet muistavat, joystickin valinnasta riitti keskustelua välitunneilla ja kavereiden kanssa iltaa viettäessä. Jonkun mielestä Quickshot II turbo oli paras, seuraava arvosti TAC-2:ta, kolmas piti Terminatoria kaikkein parhaana.*

**E**nsimmäiset omat joystickini olivat vuonna 1986 Commodore 64:n mukana tulleet Commodore VG-115:t. Alun perin Beeshun valmistamat muoviset rimpulat olivat tuntumaltaan melkoisen epämääräiset, ja kalvokytkinten päällä olleet muovitarat irtoilivat tämän tästä. Silloin pelin jatkuminen edellytti huolellista uudelleenteippausta. Tästä huolimatta niilläkin veivattiin vuosikaudet.

Vuonna 1990 löysin kotikaupunkini Info-kirjakaupassa käydessäni lattialta laatikon, josta sai ostaa rikkinäisiä joystickejä 20 markalla kappale (nyky-

rahassa 5 €). Kävimme sieltä porukalla aina hakemassa muutaman, joista sitten lapsuuden innolla juottelimme kasaan yhden toimivan. Korjattavat tikut olivat useimmiten avoimin mikrokytkimien varustettuja Commodore VG-200:ia (Cheetah/Beeshu-tuote) ja erilaisia mikrokytkettyjä Quickshot-eja. Näillä pelituntuma parani jo hie- man, mutta luotettavuus oli vieläkin aika kehno, sillä tikkujen kytkimiä sai korjata melko pian uudestaan. Muoviosien murruttua tikku lensi aina va- raosakasaan ja kaupasta haettiin uusi kytkinviallinen tilalle korjattavaksi.

### Harharetki TAC-2:n omistajaksi

Mikrobitin joystick-testin ylisanojen inspiroimana aloin himoitsemaan TAC-2:ta. Sainkin lopulta sellaisen ruinatuna vanhemiltani, mutta pettymys oli todella karvas. Vaikka tikun suunnat toimivat aina todella luotetta- vasti, tulitusnappulat jättivät harva se kerta kytkemättä, ja hermot menivät, kun elämiä kului pelivälineen epä- luotettavuuden takia. Napit toimivat aina hetken, kun olin viilannut niiden alla olevat messinkiset prikat kiiltäviki- si, mutta ne hapettuivat takaisin pätki- viksi melko nopeasti.

Todella turhautuneena porasin TAC:n kuoreen reiän ja asensin siihen elektroniikkamyymälästä ostamani painokytkimen. Paikka ei kuitenkaan ollut kovin ergonominen ja niinpä siirsin TAC-2:n hyllyyn ja palasin avoimiin mikrokytkimiin QuickJoy Superboardin muodossa. Kytkinten vaihtelu tuli taas tutuksi – olihan siinä vaiheessa harrastusta jotenkin itses- tään selvää, että joystickin ei kuulu kestää käytössä.

Joitain myöhempiä Beeshu-tuot- teitakin pöydällä kävi, kuten lentokoneen ohjaimia jäljittelevä Zoomer, joka osoittautui täysin kelvottomaksi mihinkään käyttötarkoitukseen. Ultimate Superstick taas huvittavan su- perlatiivisesta nimestään huolimatta oli yllättävän mukava tikku – ainakin niin kauan, kuin sen kaukosäätimen nappuloita muistuttava kytkinmekani- smi kesti ehjänä.

### Ensimmäinen Competition Pro

Elettiin vuotta 1993 ja olin perheeni kanssa käymässä Englannissa kesä- lomalla. Lontoon Piccadilly Circusin HMV-myyvälässä näin hyllyllä vih- reän Competition Pro Starin. Tikku kiehtoi minua, sillä monissa peleissä valikoiden grafiikoiden joystick oli ni-





menomaan Competition Pron muotoiseksi piirretty, ja olin myös arvosteluita ymmärtänyt, että kyseessä voisi olla ihan pätevä tikku. Minulla oli mukana riittävästi rahaa, jos malttaisın olla ostamatta musiikki-CD:itä, ja niinpä poistuin myymälästä paketti kainalossa ja aloin odottelemaan malttamattomana kotimatkaa.

Kotiin päästyäni tikku ylitti odotukseni. Tuntuma oli todella hyvä, joskin nörttilapsen käsille teräksinen palautusjousi oli hieman liian jäykkä ensi alkuun. Minua miellyttivät tikun lyhyt liikerata ja teollisuudessa rajakytkiminä käytetyt suljetut mikrokytkimet. Kaikki ilmansuunnat löytyivät tarkasti, eikä missään ollut epämääräisyyttä.

Tämän jälkeen en enää muihin tikkuihin juuri katsonut. Toki erilaisia jä-mätikkuja oli nurkissa, että pystyimme kavereiden kanssa pelaamaan kaksinpelejä, mutta yksin pelailessani väänsin aina Competition Prolla.

Tikun kestävyudesta kertoo jotain se, että minulla on tuo sama lapsena ostamani yksilö yhä tallessa, ja näiden 24 vuoden aikana siihen on pitänyt vaihtaa ainoastaan kaapeli ja yksi mikrokytkin. Sittemmin olen päästänyt tuon lapsuuden tikkuni eläkkeelle, mutta vielä nykyäänkin aina pelatessani käytän jotain Competition Pro -tikuistani.

## Sokka irtoa

Jossain vaiheessa 2000-lukua haaveilin ostavani vihreälle tikulleni kaveriksi samanlaisen mutta sinisen. Tarjontaa selailessani huomasin, että näitähän on tehty aika paljon erilaisia.

Alun perin tikkuja tehtiin mustana ja läpinäkyvänä punaisin napein. Kahvan palauttimena toimi varren ympärille pujotettu paksu kumikiekko. Ajan myötä mukaan tulivat erilaiset erikoismallit, kuten vihreä ja punainen läpinäkyvä, ja vuonna 1990 jalkapallon MM-kisojen innoittamana tikkuja sai osallistujamaiden lippujen väreissä. Loppupään tikuissa kumikiekko korvattiin teräksisellä jousella.

Hamsterigeenini aktivoituivat, ja lempitikkuni alkoi näyttämään silmis-



Ensimmäinen keräämäni Limited.

säni potentiaaliselta keräilykohteelta. Vuosien mittaan tulinkin sitten hankineeksi Ebaystä ja eri foorumeiden kautta melko kattavan kokoelman erilaisia värivaihtoehtoja.

Kun otin valikoimasta selvää ja katselin eri mallien kuvia myyntilaittikoiden kyljistä ja tietokonelehtien vanhoista mainoksista, alkoi mieleen hiipiä epäily, että oliko sittenkin tullut haukattua liian iso pala purtavaksi. Kaikkien eri kumijousisten väriyhdistelmien löytäminen kokoelmaan alkoi vaikuttaa melko mahdottomalta.

Lisäksi olin huomannut, että vanhojen mallien kumikiekko oli yksilöstä riippuen joko aivan liian jäykäksi kovettunut tai todella väljäksi kulunut. Vanhempien yksilöiden tapauskohtaisesti vaihtelevan pelituntuman vuoksi olin itsekin valikoimut aktiiviseen käyttöön käytännössä pelkästään teräs-jousisia tikkuja, kumijousiset olivat vain keräilykappaleina hyllyssä.

Jo aiemmin olin huomannut, että lapsena ostamassani tikussa ei ollut muoviosien karhennuksena perinteistä nahan pintaa jäljittelevää satunnaiskuvioita, vaan tikun jalustan kuvio muodostui sadoista pienistä Competition Pro -silueteista. Päätin, että jatkossa keskityn pelkästään sellaisiin teräs-jousisiin malleihin, joista löytyy tämä söpö joystick-kuvio muovin pin-

nasta, jotta kokoelmani laajuus pysyisi jotenkin tolkullisena. Moin pikkuhiljaa kaikki kumijousiset yksilöt pois ja siirsin kaikki perinteisemmällä kuvioilla karhennetut tikut aktiiviseen pelikäyttöön. Joystick-kuvioituidun jäädytin keräilykappaleiksi.

## Yksisarvisen metsästy

Vaikka olinkin päättänyt rajoittaa keräilyni vain tietyllä muotilla valettuihin tikkuihin, haastetta on silti piisannut, ja muutama harvinaisuus on vielä haalimatta. Ystäväni ja kanssakeräilijäni Antti Kiurun kanssa huomasimme joitain vuosia sitten Ebayssä erikoisia Limited Edition -tikkuja. Kävi ilmi, että vuonna 1993 tikkujen valmistaja Dynamics Marketing päätti Competition Pron kymmenvuotisen taipaaleen kunniaksi julkaista neljä erikoismallia, joita valmistettiin 9999 kappaletta. Olin onnistunut haalimaan itselleni niistä kirjavasti kuvioituidun jo aiemmin, ja Antilta sain vaihtokaupalla mustan. Löysin hiljattain Saksasta vaalean version, mutta kaikista harvinaisin, salamakuvioitu limited on itseltäni vielä löytämättä. Salamakuvioituidun lisäksi nykytiedon mukaan kokoelmastani puuttuvat enää läpinäkyvät savunharmaat sinisillä ja harmailla kolmionapeilla. Eivätköhän nekin vielä joskus löydy. 🐾







Kekkonen, Keke ja skene

## IISALMELAINEN GRAFFITI vaalii digitaalista kulttuuriperintöä

*Suomalaisen demoskenen alkulähteiltä löytyy Iisalmi ja Byterapers. Kaupunki antoi ikuistaa paikallisesta toiminnasta maailmalle ponnistaneen digitaalisen vaikuttajan ikoniseen sijaintiin.*

Teksti: Suvi Sivulainen

Kuvat: Riikka-Liisa Kajanus, Byterapers, Grendel/Byterapers

**A**lkuvuodesta 2016 Digitoday uutisoi, kuinka ”Iisalmi kunniottaa suomalaista tietokonetaidetta graffitissaan”. Samasta graffitista uutisoi myös Iisalmen Sanomat edellisenä kesänä, kun maalaus oli vielä suunnitteilla.

Mutta miksi ja miten demoskeneryhmä **Byterapers** päätyi tietämättään mukaan suomalaisen pikkukaupungin alikulkusillan graffitiin? Näihin kysymyksiin ei aiemmissa artikkeleissa tai Byteillä itsellään ollut vastausta, mutta nyt etsintä on tuottanut tulosta. Tekijä on löytynyt.

Vastapäättä kahvipöydän äärellä istuu vaalea nainen, joka puhuu innoissaan

Iisalmesta ja sen kulttuurielämästä. Hänen mukaansa kaupunki on täynnä luovia ja kiinnostavia ihmisiä, minkä vuoksi sieltä on syntynyt maailmalle paljon hyviä juttuja. Yksi niistä on tietokonetaiteen alakulttuurissa tunnettu demoskeneryhmä Byterapers.

Graffitin takana on **Riikka-Liisa Kajanus**, Turun yliopiston maisematutkimuksen opiskelija ja mediakasvattaja, joka työskenteli viime vuosikymmenen vaihteessa MySpacella El Segundon toimistossa, Yhdysvaltojen länsirannikolla, ja muutti 2011 takaisin Suomeen. Hän on Byterapersin kanssa samalta paikkakunnalta, lähes samaa ikäluokkaa ja kertoo jollain etäisellä tasolla tienneensä ryhmästä jo pitkään.

### Julkista taidetta katukuvaan

”Graffitihan sijaitsee Iisalmen pääväylällä, alikulkusillassa josta pääsee kaupunkiin.” Se on sama seinä, jolle joskus 1990-luvulla ilmestyi teksti ”Nolou, the best rappers in town”, Kajanus selittää. ”Se oli minun sukupolvelteni sellainen maamerkki: aina kun tuli Iisalmeen, näki ensimmäiseksi sen. Siksi halusin pääväylälle jotain erityistä.”

Väylä oli keskeinen myös kaupungin mielestä, jonka luvalla ja jonka kans-

sa yhteistyössä graffitit tehtiin. Vaikka teoksia tehtiin useampi eri puolille kaupunkia, niin juuri tuon paikan osalta kaupungin teknisen viraston kanssa käytiin pitkiä keskusteluja siitä, saadaanko seinät edes käyttöön. ”Jälkeenpäin ei ole tullut mitään valituksia”, Kajanus toteaa hiukan huojentuneen oloisena.

Graffitin tarina alkoi, kun Kajanus oli pitämässä valokuvakurssia Iisalmen kuvataidekoululle muutama vuosi sitten. Yksi oppilaista, 13-vuotias **Veera**, esitteli kuvaamiaan kohteita, harmaita seinä hakaristeineen ja esitti ratkaisevan kysymyksen: ”Miksi Iisalmessa ei ole kuin töhryjä?” Se sai valokuvausohjeittajana toimineen paluumuuttajan pohtimaan ympäristöään uudella tavalla ja tarttumaan toimeen. ”Kaupunki oli aika ankea. Halusin siihen muutosta.”

Riikka-Liisa kertoo olleen alusta asti selvää, että kuvaan haluttaisiin monenlaisia asioita, joista Iisalmi voisi olla ylpeä. ”Lähetin tammikuussa 2015 ekat kuvat paikasta graffitin toteuttaneelle ryhmälle.” Alkuperäinen idea oli maalata toiselle puolelle tunnelia ihmemaa Ozin kohtaus, jossa tie johdattaa joukon kohti kaupunkia – aivan kuten tunnelin läpäisevä tie vie ihmiset Iisalmen keskustaan. Kuva olisi sisältänyt erilaisia merkkihen-



kilöitä, ja toisella puolella olisivat yhdistyneet kaupungin siluetti ja nimi.

## Graffitin toteutus

Suunnitelma muuttui ja hioutui eri tahojen yhteistyönä, mutta tärkeimmät piirteet alkuperäisestä ideasta, eli kaupungin siluetti sekä Byterapers- ja Nolou-tekstit jäivät mukaan lopulliseen kuvaan. Kesällä 2015 Iisalmen kaupunki hyväksyi suunnitelman ja sen jälkeen päästiin toteutukseen.

Graffitin maalasi kolmihenkinen **Bo-Axel**-ryhmä. Itse kuva-aiheen tarkka sisältö hioutui ja tarkentui pikkuhiljaa prosessin edetessä. Vaikka kyseessä on graffiti, siinä on yksi Byterapersiin liittyvä tekninen erityispiirre: Byterapers-tekstin mustaa väriä on maalattu osin myös siveltimellä, vaikka kaikki muu on tehty spraykannuilla.

Tekijät myös havahtuivat jossain vaiheessa maalausta siihen, että ohi kulkevat lapset jäivät tavaamaan tekstin kohtaa ”rapers”. Koska konteksti ei välttämättä avaudu kaikille aivan oikein, pelasi ryhmä varman päälle piilottamalla osan tekstistä taktisesti sijoitettujen pensaiden taa.

Byterapers-nimi valikoitui demoskenen alkuaikoina, kun toimittiin vielä hyvin rajallisella prosessoriteholla toimivilla tietokoneilla. Saadakseen demoja aikaiseksi oli ohjelmoinnissa käytettävä erityisen paljon luovuutta ja erilaisia jippoja. Nimi viittaa hakkereihin, jotka pyrkivät ylittämään laitteiden tekniset rajat koodinsa avulla.

Kehun Bytejunan päällä olevia lentäviä diskettejä ja bittejä ja Kajanus kertoo niiden olevan täysin graffitin tekijöiden käsialaa. Maalaus ja sen elävät yksityiskohdat ovat saaneet muiltakin positiivista palautetta. Esimerkiksi Pariisin kevät -yhtyeen kosketinsoittaja Ilari Kivelä on ylistänyt Facebookissa

graffitia Byterapersin saaman huomiota vuoksi. ”Vaikkei ole edes Iisalmenta”, tarkoittaa Kajanus.

”Tietenkin tällaisissa projekteissa joutuu miettimään taiteen asemaa, mikä on se taide siinä, mutta teos kertoo kaupungin historiasta ja potentiaalista”, toteaa Riikka-Liisa ja jatkaa pohdintaansa siitä, voiko julkisella taiteella vaikuttaa myös yleiseen ilmapiiiriin. ”Ehkä voi, koska se on levittäytynyt yleiseen keskusteluun, lehtijuttuihin ja niin edelleen. Sellainen kiva työväline.”

## Byterapers mukaan graffitiin

Mutta miten Byterapers lopulta päätyi kuvaan? ”Se oli monen tekijän summa”, toteaa Kajanus. Tärkeää oli Iisalmen kulttuuriperintö, johon Byterapers kuului keskeisesti. ”Nämä ovat meidän 1980–90-luvun historiaamme, jota pitää ylläpitää ja muistaa.” Seinän legendaarinen Nolou-teksti haluttiin säästää, mutta Kajanus halusi mukaan myös muita alueen erityispiirteitä.

”Kuten kerroin, Byterapers-ryhmä oli periaatteessa jo jossain määrin tiedossani.” Tieto graffitia varten tyhjennetyistä seinästä levisi sosiaalisessa mediassa, ennen kuin graffitihankkeesta tiedettiin laajemmin. Moni paikallinen kaipasi poistettua Nolou-tekstiä. Samassa ketjussa yksi kirjoittajista myös totesi, että kaikki ”Byterapers-logotkin on jo taidettu hävittää kaupungista, prkl!”

Tilaisuus oli liian hyvä jätettäväksi käyttämättä. Kajanus kertoo käyneensä lukemassa Byterapersista kertovaa wikipediasivua ja ryhmän verkkosivua. ”Se verkkosivu oli negatiivina. Mustalla pohjalla valkoinen teksti!”. Kajanus myös tutki YouTubeista Byterapersin tuotoksia, kuten demoja ja ryhmän toimintaa muistelevaa videota. ”Kyse on ihmisten tekemisistä, kult-

## BYTERAPERS

Byterapers on yksi maailman vanhimista edelleen toimivista demoryhmistä. Se on perustettu 1986 Iisalmissa, ja ryhmä osallistuu yhä audiovisuaalisilla ohjelmillaan eli introilla ja demoilla kilpailuihin Suomessa ja muualla Euroopassa.

Byterapers järjesti 1980–1990-lukujen vaihteessa useita demokilpailuja ja oli näin mukana luomassa demokilpailujen nykyaikaista formaattia, jossa kilpailun voittaja muotoutuu tapahtumassa paikalla olevien äänestyksen tuloksena.

Byterapers oli myös ensimmäisten joukossa järjestämässä suomalaisia demotapahtumia. Demopartyja on yli-päätään Suomessa järjestetty vilkkaasti, ja tapahtumiin saapuu usein osallistujia myös muista maista. Partyjen rinnakkais-tapahtumina voi nähdä LAN-partyt ja erilaiset pelifestivaalit.

Demoskenestä saa lisätietoa vaikkapa Ylen vuonna 2010 verkkoon tuottamasta seitsemänosaisesta Demoskene-dokumentista: [areena.yle.fi/1-1295277](http://areena.yle.fi/1-1295277)

tuurisesta toimimisesta ja paikallisesta perinnöstä. Eniten ehkä iski demoskenen yhteisöllisyys”, hän selittää.

Graffitissa on mukana myös esimerkiksi **Kekkonen, Spede** sekä **Keke Rosberg**, joka formulauransa alussa ajoi Iisalmen kauppahallin takana olevalla mikroautoradalla. ”Rosberg ei ole tietävästi käynyt katsomassa tätä. Odotan innolla mitä Keke sanoisi tästä teoksesta.”

Maalausprosessin aikana myös Nolou-throw upin mysteeri paljastui, kun alkuperäiseksi taiteilijaksi esittäytynyt henkilö tuli kertomaan tekstin taustoja. Ilmeisesti näitä Iisalmen parhaita räppäreitä ei koskaan ollut olemassa, mutta teksti on kulttuurihistoriaa joka tapauksessa. ”Kekkonen oli suunnilleen yhtä pitkään vallassa kuin Nolou, tosin graffitissa Noloun betoni murtuu”, Kajanus vertailee.

Kajanus harmittelee, ettei heidän tempauksensa ole vielä aktivoitunut muita ryhmiä samankaltaiseen digitaalisen kulttuuriperinnön kunnioitukseen. ”Joku voisi tulla tekemään vaikka pelikulttuurista samantapaisen jutun – se olisi hyvä meidän yhteisöllemme.”



Graffiti tekovaiheessaan. Tässä kohtaa logon loppuosa on vielä sensuroimatta.



Alkuperäinen Nolou-graffiti, johon myös uudessa työssä viitataan.

## Retrointoa ja sukupolvinostalgiaa

Jatkamme vielä hetken omien 1980- ja 90-lukujen kokemustemme kertaamista ja mietimme teknologiaan liittyvää kulttuuriperintöä ja retrointoa. ”Nythän on tulossa muun muassa Commodore 64 Mini. Ehkä tuo 80–90-lukujen kulttuuri nousee uudelleen esille nyt. Johtuu ko se ikäryhmän nostalgista vai mistä, mutta se tuntuisi olevan jotenkin pinnalla. Vähän kuin itselläkin,” jatkaa Kajanus.

Kajanuksen tietokoneharrastus oli astetta yksinäisempää. Hän kertoo pelanneensa kasetilta ladattavia pelejä omalla Commodore 64:llään, mutta hänellä ei ollut tuohon aikaan tietokoneista kiinnostuneita ystäviä. Koodaamistakin hän olisi aikoinaan halunnut opetella, mutta kun kokeilu ei tuottanut tulosta, harrastus jäi.

”Ostin Savonkadun kirjakaupasta koodauskirjan ja yritin opetella sieltä”, kertoo Kajanus, mutta kun hän kirjoitti koneelle muutamia valmiita koodinpätkiä, eikä mitään tapahtunut niin kirja jäi. ”En nyt jälkeenpäin ajatellen edes tiedä oliko se kirja tarkoitettu Commodorelle”.

Myöhemmin löytyi myös muutamia tietokoneista kiinnostuneita kavereita ja Kajanus kävi pelaamassa lähellä asuvien sisarusten kanssa. ”Heillä oli Amiga 500, jolla oli parempia pelejä. Tai ainakin ne tuntuivat kehittyneemmiltä kuin ne mitä pelasin itse kasetilta. Toisaalta suhteeni omaan Commodore 64:ään oli lämmin.”

”Tulee myös mieleen miten omat polut ovat kulkeneet, kun itsellä oli aikoinaan Commodore 64, ja että olenko

ehkä missannut jonkun oman kutsukseni. Asuin nuorisotalon vieressä, mutten koskaan käynyt siellä. Myöhemmin olen tajunnut, että siellä taisi tapahtua aika hienoja juttuja. Nuorisotalon montussa ovat aloittaneet soittamisen monet muusikot ja ilmeisesti siellä oli joku nuoriso-ohjaaja, joka tuki myös Byterapersia.”

## Demoskene kulttuuriperintönä

Byterapers-ryhmään kuuluva Skrollin toimittaja **Jukka O. Kauppinen**, skenenimimerkiltään Grendel, kertoo nuorisotalon **Pete Kauppisen** kannustaneen ryhmän toimintaa. Byterapers järjesti nuorisotalolla demopartyt 1990-luvulla, ja talon yläkerrassa tehtiin vuonna 1997 The Partyilla julkaistu Yädot Klimon -demo MS-DOSille.



Yädot Klimon on esimerkki demoskenen ja laajemminkin tuon ajan tietokoneharrastamisen erityispiirteistä. Kaksi ryhmän jäsentä, toinen nimimerkiltään Sivu, vei teoksen kisaamaan Tanskassa järjestettyyn The Party -demotapahtumaan joulukuussa 1997. Demoa koodannut Sivu Byterapersista kertoo tehneensä viimeisiä korjauksia tuotokseen vielä Tanskassa, partyjen tapahtumapaikalla.

Suomessa demoskenen toiminta on erityisen aktiivista. Luova musiikin, kuvan ja ohjelmointitaitojen kautta kilvoittelu ja uuden luominen omasta halusta, ilman kaupallisia tai muuten laajasti tunnustettuja päämääriä on erittäin suomalaista. Tietotekniikkaan liittyvä kulttuurinen kehitys on kuitenkin niin nopeaa, että mikäli kentällä tapahtunutta ei tallenneta ja aktiivisesti muisteta, voi olla hyvin vaikea enää suhteellisen lyhyenkään ajan päästä kunnollisesti ymmärtää alan historiallista kehitystä, toimintatapojen syitä ja tapahtuneita muutoksia.

Demoskenessä heijastuu teknologian ennakkoluuloton käyttö ja uuden kehittäminen, jotka voi nähdä monin tavoin pioneiryönä tietokonekulttuurin saralla. Toisaalta alakulttuureihin liittynee aina jonkinasteista sisäpiiriläisyyttä ja ehkä tätä kautta laajemmasta kulttuurisesta kontekstista unohtumista. Kajanuksen graffitiprojekti toimii tarpeellisena siltana alakulttuurin ja julkisen välillä. Se vaalii niin Iisalmen paikallista kulttuuriperinnettä kuin kansainvälistä tietokonekulttuuriakin. 🐉



# IHAN PAKKO PELATA

Kuole Rufus, kuole!

*Kuluneena syksynä pelimania, pakkomielteen ja sniiduulun vektorit kohtasivat ja jättivät henkisesti tyhjäksi.*

Teksti: Tapio Berschewsky

Kuvat: Toni Kortelahti,  
Tapio Berschewsky

**P**aheni on ostaa pelejä. Ostan pelejä Steamista, GOGista, Originista, uPlaysta, Humble Storesta, erinäisistä lunastettavia koodoja myyvistä kolmannen osapuolen nettikaupoista ja aivan erityisesti Humble Bundleina.

Ostan pelejä, vaikka en ehdi pelata edellisiä. Koska tunnistan jonkun tekijän, josta pidän. Tai tiedän yhden pelin viiden pelin ryppäystä, koska olen pelannut sitä 15 vuotta sitten ja voisin ehkä pelata uudelleen. Tai kun tuntuu naurettavan halvalta saada kymmeniä pelejä pieneen hintaan.

Steam-kirjastossani on 212 teosta. Se ei ole ihan hirveän paljon, mutta valtaosa siitä on kerääntynyt viimeisen kahden vuoden aikana sen jälkeen, kun palasin pc-pelaamiseen vuosikymmenen konsolirupeaman jälkeen. Toisaalta se on vain pintaa. Humble Bundlessa minulla nimittäin on noin 6 sivua lunastamattomia avaimia pelejä varten.

Suurin ongelma, joka tästä aiheutuu, on syyllisydentunne. En ole sen kanssa ainoa. Vähän väliä peleistä keskusteltaessa tulee vastaan ajatus ”ostan enää vain pelejä, joita aion pelata heti”. Vaihtoehtoja tuntuu olevan vain kaksi: pelien ostamisen lopettaminen tai jo ostettujen läpikäynti. Tämähän ei ole

loogista, mutta taisin jo mainita pakkomielteen.

Tästä voi ehkä arvata, että käsissäsi ei varsinaisesti ole artikkeli, vaan tämä on lähinnä hämäys, jolla pakotan itseni lukijoita hyväksikäyttämään käymään läpi pelejä pois listalta, jotta voin ostaa kaikki jouluksi julkaistavat jatko-osat suosikkisarjoihini.

## Suoraan suohon, vahingossa

Aloitin siis projektin. Aion kokeilla jokaista ikinä Steamin aleista tai Humble Bundleista ostamaani peliä, jota en ole vielä testannut. Koska heti kättelyssä tiesin, että tämä tulee olemaan raskasta, päätin aloittaa todennäköisellä suosikilla: *Deponia*.

Point'n'click-seikkailupeli Deponian kauniista grafiikasta tulee mieleen Monkey Island 3. Tuotantoarvot on monelta muultakin osin kohdallaan. Musa on kaunista. Puzzlet kohtuullisia. Olin valmis rakastamaan.

Harmi vain, että keskeinen hahmo on täysi ääliö ja pelin juonena typerä runkkufantasia taivaalta tippuvasta täydellisestä naisesta, jonka ajatuksia voi kontrolloida kaukosäätimellä.

Päähahmo Rufus, joka ei jätä läähättämäänsä naista rauhaan edes useita kertoja suoraan käskettäessä, on toksisen maskuliinisuuden oppikirja.

Toisaalta Rufus on niin monella muullakin tavalla raivostuttava, että pitäisi ehkä puhua vain yleisemmästä toksisesta paskuliinisuudesta.

Tämä ei alkanut hyvin, mutta ainakin pääsen varoittamaan heti. Älkää pelatko Deponiaa, se ei ansaitse teitä. Sääli, sillä audiovisuaalisesti näin hienoja seikkailupelejä ei näe ihan joka päivä.

## Takaiskusta huolimatta

Seuraavaksi kahlasin yhdessä viikonlopussaläpi *Spacebase DF-9:n*, *Tiny Bridge: Ratventuren* ja *Project Highrisen*. Vähän ohuita pelejä näin, mutta ainakaan rakentelu ei ketuttanut kertakaan.

*Okhlos: Omega* taas jätti jäätävän kylmäksi. En tule ikinä käynnistämään sitä toiste, vaikka antiikin Kreikka on sitä historiaa, jolle sydämeni sykkii eniten.

Jatkoin antiikissa *7 Grand Steps, Step 1: What Ancients Begatin* kanssa. Nimeä on riittämiin, mutta lautapeli-mäistä kokemusta lopulta aika vähän. Luotsasin silti sukuani savimajoista herruuteen yhden yön ajan ja olin seuraavana päivänä väsynyt töissä. Krokotiilit söivät useita serkkuja.

Yhden syyskuisen vapaapäivän aikana testasin putkeen *A Virus Named*





Tunnelmallisin peli hyvin pitkään aikaan. *What Remains of Edith Finch*ä on helppo suositella kaikille. Aina voi sitten vihata niitä, jotka ei tykkää.

*TOMia*, *Girls Like Robotsia*, *Beat Hazardia*, *Dusty Revengeä*, *Ellipsisiä*, *Flat Heroesia* ja *Expandia*. Näistä ei ole yhtään mitään sanottavaa. Jälkikäteen en edes muista mitä pelejä *Beat Hazard* tai *Dusty Revenge* olivat katsomatta screenshotteja. Suoraan sanoen niistäkin vähän huonosti. Sen jälkeen rajoitin uudet pelit yhteen per päivä, jotta en tuupertuisi matkalla.

### Väliaika, jolloin on kivaa

Tässä välissä luovutin hetkeksi ja ostin *Life Is Strange: Beyond The Stormin* sekä pelasin ensimmäisen episodin. Ah, niin ihanaa. Saisipa useammassa pelissä pelata angstista teiniä. Muutenkin kuin japsiroolipeissa siis. En yleensä jaksa japsiropeja.

### Jatketaan kävellen

*Life Is Strangen* ihanuuden takia päätin jatkaa listalta löytyvien kävelysimulaattorien parissa. Melkein kaikissa peleissä nimittäin on aivan liikaa itseään toistavaa mättöä. Näissä ei.

Lähdin liikkeelle *Firewatchilla*. Kuumottava tarina palovartija-amatöörin ensimmäisistä työkuukausista on niin kova, että sen pelaamatta jättämisen pitäisi olla lakisääteisesti validi syy kickbannata ihmisiä irkkikanavilta.

Tosin sitä seuranneeseen *What Remains of Edith Finchin* verrattuna

*Firewatch* oli kevyttä kamaa. Ihastuttavilla ja puistattavilla tavoilla kiero *What Remains of Edith Finch* on vuoden kevyesti paras ja yllättävin peli ja ehkä *Witcher 3:n* jälkeen paras tarinallinen peli tältä vuosikymmeneltä. Olen puhunut.

Seuraavaksi testiin tuli *The Beginner's Guide*. Ajatuksia herättäneessä teoksessa kuljetaan läpi fiktiivisen amatöörinkehittäjän taidepeleistä tehdystä koosteesta. Peli ihastutti ajoittain aika rasittavaksi äityvästä pseudofilosofisista pelinkehitysjuuttuja selittävästä kertojasta huolimatta. Tämä oli jäätävän ruma, teennäinen, kaunis ja hieno yhtäaikaan. Ei missään nimessä samassa laatuluokassa kuin *Firewatch* tai *Edith Finch*, mutta kannatti pelata.

Kokeilin myös viimeinkin suomalaista *INFRAA*, jossa kuljetaan fiktiivisessä keskieuropalaisessa kaupungissa korjaamassa budjettileikkausten seurauksena rapistuvaa infrastruktuuria. Teos on hieno, poikkeuksellinen ja suositeltava – muutenkin kuin suomalaisuuden vuoksi. Lisäksi siinä puhutaan suorastaan hulvatonta rallienglantia. ”Ei bäk öv sbiör bädöriis” on nyt vakioheittoja kotona, kun jokin laite tarvitsee uudet paristot.

Näiden jälkeen rentouduin hetken *Everythingin* parissa. *Everythingissä* pelataan nimensä mukaan kaikkea

mahdollista. Välillä ohjataan alkuhiukkasia, välillä kamelia, välillä lintua ja välillä galaksia. Samalla kuunnellaan taustalla iki-ihanan Alan Wattsin hypnoottista pohdiskelua olemisen todellisesta luonteesta. Ehdotan, että alasteen opetussuunnitelmaan lisätään joka aamuun ensin tunti pikkukivien järjestelyä värin ja koon mukaan ja sen jälkeen puolen tunnin sessio *Everytingin*. Olisimme kansakuntana terveempiä.

### Toinen väliaika

Lokakuun lopussa julkaistiin toinen episodi *Life Is Strange: Beyond The Stormiin*. Sori! Ihan pieni tauko vain!

### Pääosin toissijaista hyppeilyä

Tasohyppeilyjen aika. Minä, kuten ehkä joku muukin, turhauduin kovaa *Rick Dangerousiin* lapsena. Siinä nimittäin kuoli tosi tosi paljon tosi tosi tosi turhauttaviin ansoihin.

No, joku oli ajatellut, että on hyvä idea kloonata *Rick*, mutta poistaa puolet ominaisuuksista. *1001 Spikes* on *Rick Dangerous* -klooni jopa niin pitkälle, että päähahmo näyttää *Rickiltä*. Tai siis, no. Näyttäisi *Rickiltä*, mutta on rumempi ja yksinkertaisempi verrattuna esikuviinsa, kuten nyt keskimäärin pikseligraffa indiepeleissä ylipäätään.

*1001 Spikes* vaatii tietynlaista maso-



kismia pelaajalta. Pelidesign pyörii Rickin tapaan sen ympärillä, että kahden sekunnin välein joku uusi epäreilu ansa tulee seinästä, lattiasta, katosta tai jostain muualta ja tappaa. Äkkikuoleman määrästä tulee mieleen VVVVVV:n huikea Veni, Vidi, Vici -kohta, jota taisin aikanaan yrittää noin tuhat kertaa ennen ensimmäistä onnistumista. Olisipa uusi VVVVVV.

*Slime-sanissa* taas hyppii limapallero. Vähän siis Gishin tapaan, mutta paljon huonommin. Kentät ovat tylsiä, grafiatyylillä omaperäisellä tavalla rumaa, ja unohdin tästä oikeastaan kaiken muun heti quitattuani.

*Ninja Pizza Girlsissä* jaksoin viilettää muutamanakin iltana. Pizza on pakko saada äkkiä perille, ja siihen paras keino on tunnetusti lähettää parkourpizzaninja juoksemaan kaupungin katujen halki tomaattilätty kainalossaan.

Eihän tämä mikään huippulaadukas ollut, mutta tässä vaiheessa projektia kohtuullinenkin tuntuu kavalta, ja toisaalta vauhtia pelissä kyllä riittää. Sujuvat juoksut kujien, kattojen ja parvekkeiden läpi tuntuivat tyydyttäviltä, ja pizza nyt on aina pizzaa.

## Kosmokseen ja mikrokosmukseen

Marraskuussa lähdin avaruuteen *Walking Marsin* myötä. Erikoinen ja nättillä tavalla ruma tasohyppelypuzzleiseikkailu, jossa kasvatetaan Marsin uumenissa outoja kasveja, ei jaksanut kiinnostaa sekään yli tuntia.

Seuraavaksi kokeilin sitä toista avaruutta, jossa asuu höyrybotteja. Ruotsalaisten keksimä *Steamworld Heist* on aika hauska sivulta kuvattu vuoro-

pohjainen strategia. Sekään ei kannata montaa iltaa, mutta muutaman taistelun ajan oli erittäin kivaa. Pelasin sen innostamana myös täysin eri genreä olevan *Steamworld Digin*, joka oli kaksikosta parempi. Tässä on se huono puoli, että nyt tekee mieli ostaa juuri julkaistu *Steamworld Dig 2*. En kuitenkaan anna periksi.

Kutsuin sen sijaan erään hakkerin kylään ja pistimme kolmistaan tulille polyamorisen *Lovers in a Dangerous Spacetime*, jossa lähdetään pelastamaan pikkuutuksia ja palauttamaan rakkautta galaksiin isolla yhdessä ohjattavalla aluksella. Kaksistaan se ei oikein suju, mutta kolmella pelaajalla oli oikein mukavaa. Neljällä varmasti vielä parempaa, kunhan kaikki ovat sopivan samanhenkisiä. Parhaita yhteistyömoninpelisekoiluja hetkeen.

Yritin pelata myös *Morphopolista*, joka on ainakin videon perusteella täynnä yksityiskohtaista grafiikkaa kasveista ja ötököistä. Se tosin ei suosunut käynnistymään oikein, vaan tarjosi aina mustaa ruutua ja sekoitti näytön resoluutioasetukset, kun sen pakotti sammumaan. Sikäli kokemus on hyvin autenttinen simulaatio pc-pelaamisesta Windows XP -ajalta, mutta en innostunut siitä. Enkä pääsyt kokeilemaan itse teosta.

## Ei loppua näkyvissä

Syksyn aikana ehdin purkaa jonoa yhteensä 32 pelin verran, ja niistä muutama olivat niin surkeita teoksia, etteivät ansaitse edes mainintaa tässä. Mikä on siis jo aikamoista laatua, kun kerroin kuitenkin sekä Deponiasta että pelistä, joka ei käynnistynyt lainkaan.

## Koodeja halukkaille

Humble Bundleja ostellessa ei voi helposti välttää duplikaateilta. Alla siis muutama ihka ilmainen Steam-koodi lunastettavaksi. Koodit voi käyttää vain kerran, joten kukin menee sille Skrollin tilaajalle, joka ehtii yrittää ensin. Tämä on ikään kuin maksu siitä, että teidän ansiostanne sain kasaa vähän raivattua. Jos satut olemaan se ekana lehden käsiinsä saanut, niin ole reilu ja lunasta vain yksi. Kiitos. <3

ABZU  
G8XPB-57H3T-D4FV5

Everything  
89NYL-TLFFJ8-3QYYI

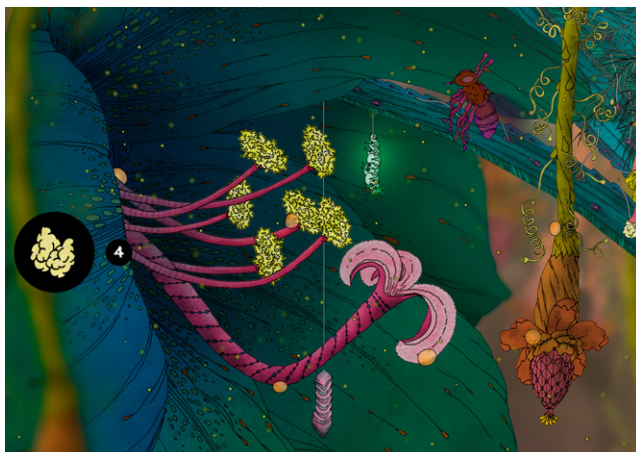
GRAV  
69TZ0-9NMPA-KAQN9

Human Resource Machine  
JEMN9-44QX6-FXEDA

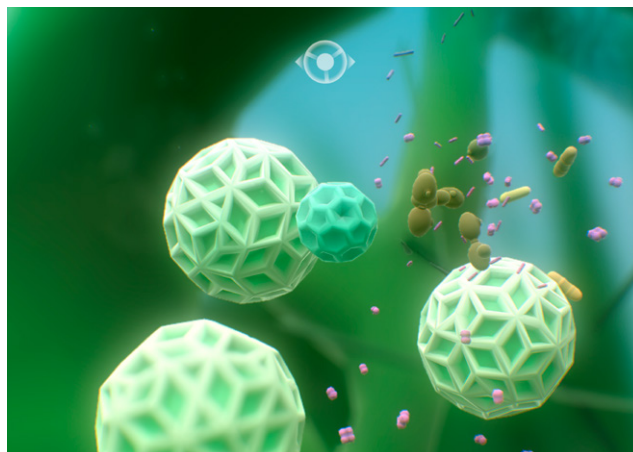
Mushroom 11  
PWR9C-W5TM5-WBMGF

Kokeiltavaa on jäljellä vielä yli sata nimikettä. Talvesta on tulossa aivoa raastavan pitkä. Seuraavaksi listalla ovat *Event[o]*, *Knight Squad* ja *Rituals*. Paneudun niihin ihan kohtaa.

Annan kuitenkin itselleni nyt oikeuden yhden pelin ostamiseen täksi jouluksi, eli *Assassin's Creed Originsin*. Kuka nyt ei haluaisi kiivetä Gizan pyramidien huipulle mukavasti kotisohvalta. Sitä paitsi tarvitsen terapiana huikeaa grafiikkaa, sujuvaa ohjausta ja hiottua mekaniikkaa. Ansaitsen sen. 🎮



*Morphopolis* näyttää hyvältä. Tai näyttäisi, jos se latautuisi. Omalla koneella ei. Harmi sikäli.



*Everything* on ainoa peli ikinä, jossa olen pelannut siitepölyä ja hengannut yhdessä kamusiitepölyhiukkasten kanssa.



# Game Music Collective

## – pillit ja kellot, bitit ja sellot

*Suomessa tehtiin syksyllä 2017 historiaa. Euroopan ensimmäinen ammattilaisista koostuva pelimusiikkia soittava yli 20 hengen orkesteri soitti debyyttikonserttinsa Helsingin Finlandia-talolla.*

Teksti: Tapio Berschewsky

Kuvat: Game Music Collective

**E**simakua konsertista saatiin, kun Game Music Collectiven viisi sellistä yhdessä beatboxaaja **Rudi Rokin** kanssa esittivät Assembly-tietokonefestivaalin avajaisissa Skyrimin pääteeman sekä Game of Thrones -sikermän.

Varsinainen idea orkesterista syntyi jo syyskuussa 2016. **Lukas Stasevskij** oli YouTubea ja Spotifyä käyttäessään huomannut pelimusiikin suuren suosion. Videoita ja kappaleita oli pilvin pimein, ja soittokertoja riitti. Jotain kuitenkin tuntui puuttuvan.

”Kaikelle muulle musiikille oli omat orkesterinsa, mutta pelimusiikille ei. Persoonallisen, mielikuvituksellisen ja hyvin monipuolisen pelimusiikin esittäminen vaatii korkean ammattitaidon. Kipinä oman pelimusiikkiin eri-

koistuneen orkesterin perustamiseen syttyi”, Stasevskij muistelee. Yleisölle tulisi tarjota paras mahdollinen pelimusiikkielämys.

### Alkusoitto

Orkesterin taiteellinen johtaja Stasevskij löysi vastakaikua muusikoissa ja rakensi yhdessä manageri **Katariina Nybergin** kanssa Euroopan ensimmäisen ison, ammattimaisen pelimusiikkiorkesterin. Mukaan saatiin palkittu nuori kapellimestari **Eero Lehtimäki**.

Ensimmäistä kertaa orkesteri soitti yhdessä huhtikuussa 2017, jolloin myös Game Music Collectiven ensimmäinen pelimusiikkivideo *Video Games Medley* julkaistiin YouTubeessa. Syyskuun konserttia edelsi erittäin intensiivinen harjoittelujakso Finlandia-talolla.

Orkesterin toiminnan mahdollistavat sponsoreina toimivat Rovio ja Veikkaus. ”On suuri kunnia saada heidät mukaan projektiimme. Sponsoreidemme kiinnostus hankettamme kohtaan kertoo myös pelimusiikki-ilmiön merkittävydestä”, Stasevskij kiittelee.

Projektin edetessä Game Music Collective lähestyi Assembly-tapahtumaa esiintymisestä. Yhdessä Assemblyjen pääjärjestäjä **Pekka Aakon** kanssa syntyi lopullinen idea avajaisista. Mainio aloitus jätti yleisön nälkäiseksi ja myös Skrollissa päätettiin heti konserttiin menemisestä.

### Debyytin tunnelmia

Konsertti järjestettiin 20. syyskuuta Finlandia-talolla. Tilaisuus oli aulasta



alkaen omanlaisensa. Näin suurta koti-housujen ja pelipaitojen keskittymää ei yleensä näe klassisen orkesterin konsertissa. Tunnelma oli odottava mutta kotoisa, mukava ja kaikkea muuta kuin jäykkä tai hienosteleva.

Yleisö oli vahvasti mukana, mutta kaikki teokset eivät olleet yhtä tuttuja kaikille. Tunnelma hieman heittelehti konsertin aikana, mutta keskittynyt ja epäuskoinen ilo koko konsertin olemassaolosta vallitsi alusta loppuun.

Konsertin juonsi hyvinkin letkeästi kapellimestari Eero Lehtimäki, joka kertoi kappaleiden väleissä peliaiheisia sukkeluuksia. Kaikesta paistoi, että lavalla oltiin lempiprojektin parissa rakkaudesta lajiin.

Musiikkia täydensi suomalaisen **VJ Sellekhanksin** hieno videotaide, jonka lavan taakse projisoidut hypnoottiset visualisoinnit saivat salin välkkymään.

## Pokeri tyhjensi pajatson

Parhaan yhteistunnelman nostatti Veikkaus-sikermä, jossa sponsorin iki-vihreät Lotto- ja Pokeri-teemat sulautuivat leikitellen toisiinsa.

Henkilökohtainen huippukohta oli kuitenkin jo alussa soitettu Monkey Islandin teema. Tippa nousi linssiin, vaikka saman sävelmän on kuullut kymmeninä erilaisina versioina sen jälkeen, kun nautti siitä ensimmäistä kertaa fm-synteesinä viiden pennin aktiivipurkeista.

On oletettavaa, että monella muulla kuulijalla oli vastaava kokemus jostain toisesta sävellyksestä. Pelimusiikissa mielikuvat parhaista kappaleista sekoittuvat erottamattomasti suloiisiin vapaisiin hetkiin lempipelin äärellä, missä niihin on tutustunut.

Tämä nostalgia yhdessä huikeiden, lennokkaiden sovitusten kanssa veivät yleisön hurmukseen. Teosten välillä taputettiin, vihellettiin, ulistiin ja huudettiin villisti kuin rokkikeikalla.

Nautin illasta suuresti. Ainoa kritiikini koskee Final Fantasy -kappaleiden määrää. Valitut teokset ovat hienoja, mutta kolmasosa koko illasta yhtä pelisarjaa tuntuu lioittelulta, kun maailmassa on niin paljon laadukasta ja monipuolista pelimusiikkia, mistä ammentaa. Skrolli toivoo tässäkin asiassa enemmän diversiteettiä tietokonekulttuuriin.

## Tulevat sävelet

Kokemuksen jälkeen vallitsi epäuskoinen tunnelma. Junaa kohti syysillassa kävellessä koko vartalo kihelmöi. Tätä pitää vielä joskus saada lisää. Annoskauteudesta kärsivät voivat suunnata Yle Areena -palveluun, jossa konsertti on melkein kokonaisuudessaan.

Seuraavaksi orkesteri suuntaa Tampereelle ja Turkuun, joissa järjestetään konsertit näillä näkymin toukokuussa 2018. Kiinnostuneet löytävät niistä ajallaan tietoa Game Music Collectiven sometileiltä ja kotisivuilta [www.weplaygamemusic.com](http://www.weplaygamemusic.com).

Konserttien lisäksi Game Music Collective aikoo myös toimia studio-orkesterina. ”Tavoitteena on tiivis yhteistyö peliyhtiöiden kanssa. Uskon, että sitä kautta pääsemme viemään pelimusiikkia entistä pidemmälle”, Stasevskij kertoo.

Skrolli toivottaa Game Music Collectivelle pitkää uraa ja menestystä. 🎮

Lukas Stasevskij halusi tarjota parhaan mahdollisen pelimusiikkikokemuksen yleisölle. Tästä halusta syntyi Game Music Collective.



## Esitetyt kappaleet

### Final Fantasy VII – Opening Theme and Bombing Mission

Säv. Nobuo Uematsu

Sov. Jonne Valtonen

### Monkey Island

Säv. Michael Land

### Undertale Medley

Säv. ToFox

Sov. Roger Wanamo

### Journey – Apotheosis

Säv. Austin Wintory

Sov. Roger Wanamo

### Final Fantasy VIII Battle Medley

Säv. Nobuo Uematsu

Sov. Jonne Valtonen

### Secret of Mana – Fear of the Heavens

Säv. Hiroki Kikuta

Sov. Roger Wanamo

### Final Fantasy Airship Medley

Säv. Nobuo Uematsu

Sov. Roger Wanamo

### Mega Man 2, 3, 4 ja 5

Säv. Takashi Tateishi, Yasuaki Fujita,

Minae Fujii, Yoko Shimomura

### Chrono Trigger Medley

Säv. Yasunori Mitsuda

Sov. Jani Laaksonen

### Veikkaus Medley – Pokeri – Lotto – Numerojytä

Säv. Kimmo Koskinen, Sakari Kuosmanen,

Matti Tuominen

Sov. Jani Laaksonen

### Angry Birds Medley

Säv. Ari Pulkkinen, Henri Sorvali,

Salla Hakkola

Sov. Jonne Valtonen

### Final Fantasy X – To Zanarkand

Säv. Nobuo Uematsu

Sov. Roger Wanamo

### Halo Main Theme

Säv. Martin O'Donnell / Michael

Salvatori

Sov. Jonne Valtonen

### Dragonborn – Main Theme from the videogame THE ELDER SCROLLS V: SKYRIM

Säv. Jeremy Soule

Sov. Roger Wanamo

### Kingdom Hearts – Destati

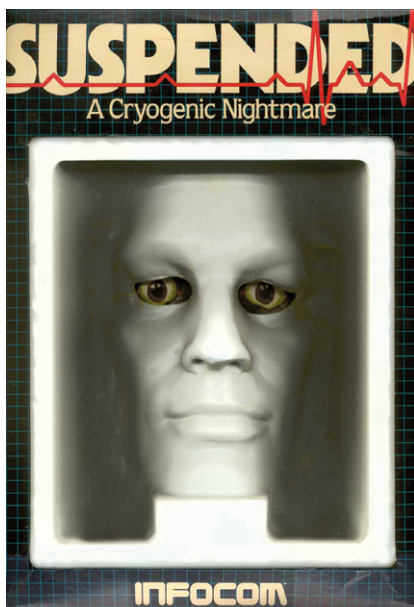
Säv. Yoko Shimomura

Sov. Roger Wanamo

Jokainen tietokone oli hämmästyttävän ja ihailun arvoinen, ja koneenomistajilla riittikin kavereita. Tietokoneharrastus oli 1980-luvun alkuvuosina ihailtavan yhteisöllistä, sillä koneiden ohjekirjoja ja vähäisiä tietokoneista kertovia lehtiä ja juttuja tutkittiin porukalla. Ohjelmalistauksia naputeltiin kimpassa ja yksinkertaisiakin Basic-merkkigrafiikkapelejä taottiin kovasti.

Eräällä tutullani oli jo varhain Sinclairin ZX81, jonka yhden kilotavun muistilla tehtiin ihmeitä. Commodoren VIC-20 vasta olikin tosi pelihirviö. Hieman oudoksun muistikuviani, mutta ihmeen moni peli taisi löytyä kavereilta ja serkuilta moduleina. Etenkin *Radar Rat Race* ja *Omega Race* ovat jääneet mieleen peleinä, joissa oli jo tuolloin sitä jotain.

VICin pelivalikoima oli aikaisekseen varsin monipuolinen. *Falcon Patrolin* räiskintä, *Flight Path 737:n* lentosimulaattorointi ja moni muukin peli herkutteli jo noina 1981–1984 välisinä vuosina monimuotoisuudellaan, vaikkakin vähän vaikeammat ja englanninkielisemmät pelit jäivät taatusti kokematta. Eihän niitä mistään saanut, eikä piraattirinkejäkään vielä ollut. Ainakin jälkikäteen on ollut kiintoisaa tarkkailla miten VICille oli myös *Donkey Kongin*, *Pac-Manin*, *Galaxianin* ja ties miten monen muun kolikkopelihatun virallisia ja etenkin virattomia versioita. Amerikan ensimmäiset tekijäoikeudenkäynnit taisivat olla vielä käymättä, ja maailma oli hyvin... sanoisinko



Teksti: Jukka O. Kauppinen  
Kuvat: Susanna Rantanen, Mobygames,  
Printti / A-Lehdet / Janne Rautiainen

*Tiedonkulkua ei voi estää. Jotenkin suuren maailman tietokoneuutuuksista kuultiin myös Kehä kolmosen pohjoispuolella, ja erilaiset koneet tulivat tutuiksi viimeistään siinä vaiheessa, kun niitä ilmaantui pikkukaupunkienkin kauppoihin.*

vapaa muiden luomien konseptien ja nimien ryöstöviljelyyn.

Jotenkin sitä sivistettiin ja päästiin parempien pelien makuun. Koulujen kerhojen kautta löytyneet tuttavuudet avasivat maailmaa, ja eräskin tosiguru oli jo 1983–1984 paikkeilla siirtymässä eteenpäin Commodore 64:stä. Melkoinen edelläkävijä, kun muut vasta haaveilivat siitä. Hänen PC-kehunsa tosin menivät aika yli, emmehän me bisneskoneista pitäneet. Hän jopa osasi englantia ja esitteli outoja, lähes käsittämättömiä Infocomin tekstiseikkailuja. Kehä kiertää, sillä törmäsin vastikään Infocomin *Suspendediin* Pelimuseon retromyyjäisissä. Sen kolmiulotteiselta näyttävä kansi ja oheiskirjääs herättivät välittömästi ihastusta.

Infocomin muinaiset tekstiseikkailut olivat huikeaa interaktiivista fiktiota, mutta harvan englannin kielen osaaminen riitti 1980-luvun alussa niiden pelaamiseen.

Harva firma panosti peleihinsä näin kokonaisvaltaisesti, edes noina menneinä vuosikymmeninä.

Infocom-seikkailut jättivät kuitenkin kutinan, jota pahensi yläasteen atk-kerhon opella ollut jännittävä suomenkielinen tekstiseikkailu. Siinä oli jonkinlaisessa kummitustalossa ja tehtiin... jotain. Pelasin sitä vain kerran, mutta se oli niin vaikuttava kokemus, että se jäi mieleen. Luulenpa, että tämä kokemus sytytti pysyvän palon seikkailupeleihin sekä ruokki osaltaan orastavaa mieltymystäni fantasiaan ja kauhuun. Myöhemmin olen kovasti yrittänyt etsiä kyseistä peliä, sillä veikkaan sen olevan erittäin varhainen, nykyään tyystin tuntematon esimerkki muinaisista kotimaisista seikkailupeleistä. Ei ole mokoma tarttunut Kasettilamereidenkaan haaviin.

### Asiaa suomeksi

Vuosikymmenen lähestyessä puoltavälään maailma muuttui vauhdilla. Koi-



visto oli uusi **Kekkonen** ja kotimikrokuume liekehti. *Tekniikan Maailman* ja *Tietokoneen* kotimikro- ja pelikonejuttuja luettiin kodeissa puhki niin innolla, että lehtitaloissakin havaittiin potentiaalinen markkinarako. *Tietokoneen* kainalossa ilmestyneestä kotimikroliitteestä syntyi *MikroBitti*, jonka kanssa kilpaili vahvasti etenkin *Printti*. Nyt tarjolla oli viimein ajantasaista tietoa tietokone maailman ihmeistä, vempeleistä, ohjelmista ja tietenkin peleistä.

Perinteinen journalismi ei muuten jyllännyt täysillä uudessa mikrolehdissä. Vaikka Suomen Kielen asian tuntijoita, taittäjia ja muuta perinteistä osaamista tarvittiin, niin datajuttuja takoiivat paljolti myös innokkaat harrastajat. Tänä onkin hauska katsoa kuinka lehdet ja kirjoittamisen tyyli ovat muovautuneet ja kehittyneet ajan myötä. Etenkin tuon ajan pelijuttuivat tuntua kömpelöiltä, mutta kyllä ne silloin mielikuvitusta kutkuttivat. Kunpa minäkin saisin tämän tikun, tuollaisen vekottimen tai tämän äimistytävän pelin! Ja mainoksillakin oli väliä.

Maailma oli myös hellyyttävästi hitaampi. Tieto liikkui mutta tihkuen. Ulkomaiset lehdet olivat matka- ja julkaisuviiveistään huolimatta tärkeä uutislähde suomalaismedioillekin, ja jos maailmalta saapui uutisia suoraan valmistajilta, niin yleensä kirjeitse. *Printti* kuitenkin sykki tietoa lähes tosijassa, viikoittain, yleensä 16–24 sivun paketeina. Painotus oli usein pienemmissä uutisissa ja lyhyemmissä artikkeleissa, ja sivuntäytettä kertyi myös listauksista ja yhdistyspalstoista, sillä tällöin elettiin vahvasti erilaisten atkerhojen kulta-aikaa. Kuukausittainen *MikroBitti* taasen loisti suuremmilla, pidemmällä artikkeleilla ja testeillä.

Pelivesteluthan muuten olivat tuolloin kovin lyhyitä. Yhdellä sivulla saattoi olla kahdesta neljään arvostelua, *Printissä* joskus reilusti enemmänkin. Toisaalta *Printissä* oli myös puolen sivun, joskus jopa koko sivun pelijuttuja, jopa tuolloin harvinaisista ilmaispeleistä.

Moni teema-artikkeli vaikutti vahvasti mikrokansan ostoskäyttäytymiseen ja harrastamiseen, minkä myötä joillakin erityisen hyvillä kirjoittajilla oli poikkeuksellisen paljon vaikutusvaltaa. Minunkin elämäni vaikutti



# PRINTTI

**TIETOKONEAJAN LEHTI BITTINIKKAREILLE HINTA 4 MK (sis. 100) 2/85**

**PÄÄTEOHJELMA**  
**C-64:ään** sivu 12

**Kun räiskintä riittää**

Seikkailupelien kirjo on laajempi kuin mitä pikaviikaisu kauppojen hyllyihin osottaa. Pelkät tekstiseikkailut ovat kehittyneet kuvalliseksi ja ottaneet mukaan räiskintä-pelien vaiheita ja sorminappäryiden. PRINTTI pelasi kymmenkunta eri seikkailupelien edustajaa lähinnä Spectrumin tarjonnasta ja syventyi klassisen tekstipelin Zorkin taustoihin.

**Kotimikrot "tilastoitu"**

PRINTTI kokosi tietoja, karsi kyseenalaista faktaa ja lajitteli mikrot neljään taulukkoon. Vertailukelpoiset ominaisuudet ovat nyt rivissä

**MSX-ryntäys käsillä**

Japanilaisten MSX-mikrojen maininnousu on näinä päivinä täyttä lottia, kun koneet alkavat olla myös kauppoissa. PRINTTI tulkaili neljää standardikonetta ja toteasi, että oman lleen kulkijalta normien ohitusta. Erityisominaisuuksiltaan kirkkain MSX-tähti on Yamaha, jonka musiikkio ominaisuudet ja siitä tukeva

**Kestääkö mikromekan maine?**

Brittiläisten tukevat imitoivat ovat huoneet mikropeleihin kovan harrastajan paratiisista. PRINTTI päätyi Lontoonen festivaalissa mieltävän todellisuutta.

**Yksäsiiv**

Räskittäminen markkassa, asiantuntija myyjä opastamassa. Maavekura Suomessa. Todellisuutta Lontoossa?

**Sivu 10**

Yamaha soi sekä näppäimistöä että lisävarusteista koskettimistoita. Stereojohdot kotivahvistimeen ja tulos on ällättävä

Vuonna 1985 mikron ostajalla riitti valinnanvaraa. Kuusnelosen lisäksi markkinoilla oli Spectrumeita, Spectravideoita, MSX:iä, Amstradeja, Oriceja ja monia tänään huonosti muistettuja, jopa unohdettuja mikromalleja. Suomenkielisille tavan kuluttajille suunnatuille tietokonelehdille oli kova tilaus, jota täyttivät muun muassa *Printti* ja *MikroBitti*.

eniten muutama aivan tietty juttu. Josain *MikroBittissä* oli näet isot artikkelit peleistä *Elite* ja *Lords of Midnight*. Jutut taisivat olla perinteisen peliarvostelun sijaan enemmänkin lukujuttumaisia reportaaseja, ja olin niin myyty. Molemmat pelit kuulostivat uskomattomilta, aivan minulle tehdyiltä. Olisipa minulla oma tietokone.

Mutta eihän minulla ollut. *MikroBittiä* ja *Printtiäkin* luin kirjastossa. Könysin kavereiden luona pelaamassa, kirjoittamassa listauksia ja näpeltämässä. Kiersin kotikaupungin mikrokaupat läpi varmasti viikoittain ja tiesin kauppiaita paremmin, mitä pelejä heillä oli myynnissä.

Kunnes eräänä päivänä. 🐱

Jukka O. Kauppinen alias Grendel/Byterapers on pitkän linjan toimittaja, joka kirjoitti ensimmäisiä tekstejään mekaanisella kirjoituskoneella. JOKin ensimmäinen peliarvostelu julkaistiin 1986, mistä lähtien hän on ahkeroinut tauotta kirjoittaen juttuja muun muassa videopeleistä, tietotekniikasta, viihteestä, ilmailusta ja burleskista. JOKistoriaa-sarjassa muistellaan niin kotimikroilun kuin peli- ja digijournalismin maailmaa grennulasien läpi nähtynä. [jukka@skrolli.fi](mailto:jukka@skrolli.fi)



# ASSEMBLYT

luovuuden  
alustana

*Assemblyt ovat jättilänit, tietokonefestarit ja paikka, jonne tullaan luomaan, tekemään ja innostumaan.*

Teksti: Heidi Akselin ja Tiina "Milsa" Malin Kuvat: Heidi Akselin

**T**ietokonefestivaali Assembly valtasi jälleen 3.–6. elokuuta Helsingin Messukeskuksen. Kesätapahtuma markkeerasi myös Assemblyjen 25. vuosipäivää, ja puitteet olivat sen mukaiset.

Järjestäjät keuhuvat kesäfestivaalinsa olevan globaalisti suurin ja arvostetuin "demoalan" tapahtuma, eivätkä keuhut tule perättöminä. Tälläkin kertaa Messukeskukseen saapui yli 8000 kävijää yli 20 maasta. Lisäksi internetin välityksellä Assemblyille osallistui noin miljoona etäkävijää ympäri maailmaa.

Perinteisesti festivaalien ohjelmiin kuuluivat muun muassa kompot, turnaukset, vapaata pelaamista jättimäisellä pelialueella, pelintekojamit sekä ammattilaisten luentoja ja yritysten esittelypisteitä. Juhlavuoden kunniaksi mediataidepuolta oli korostettu ja peliturnausten määrää oli lisätty.

## Luovuuden mekka

Messukeskuksesta sukeutui keskikesän viikonlopun ajaksi oikea modernin luovuuden mekka täynnä monipuolista ohjelmaa, pelikulttuurista vieh-

tyneitä osallistujia, alan ammattilaisia sekä kisailua. Assemblyllä on aina ollut luja sija suomalaisessa pelikulttuurissa, eikä neljännesvuosisadan ikään ehtinyt tapahtuma näytä väsymisen merkkejä.

Vaikka monet alan harrastajat saattavat mieltää Assemblyt lähinnä pelaamiseen suunnattuna viihdetapahtumana, Assembly haluaa esitellä peli- ja it-alaa myös mahdollisena urapolkuna ja tuoda alan yrityksiä lähemmäs osallistujia. Festivaali tarjoaa ainutlaatuisen tilaisuuden verkostoitua ammattilaisten kanssa ja kartuttaa omia



tietotaitojaan erityisesti pelintekemisen maailmassa.

Pelitutkija **Sonja Ängeslevä** kertoi Assemblyjen tavoitteena olevan kasvattaa osallistujien luovuutta ja synnyttää into pelien tekemiseen.

”Tärkeintä on saada se fiilis, että mäkin voin tehdä.”

Assembly haluaa tarjota helposti lähestyttävän ilmapiirin, jonka puitteissa voi kokeilla eri asioita ja löytää oman taiteenalansa. Ängeslevä korostaa, että Assemblyillä ketään ei kiusata, vaan päinvastoin rohkaistaan kokeilemaan omia siipiään. Myös testosteronivoittoisen tapahtuman naisosallistujien määrä oli lisääntynyt viiteentoista prosenttiin, mikä herätti erityistä iloa.

Ängeslevä toimii myös Assemblyjen ARTtech-seminaarin ohjelmatuottajana, jonka estradille halutaan tuoda alan kovimpia tyyppejä, innovaattoreita ja guruja kaikilta pelikehityksen osa-alueilta. 16 vuotta järjestetty seminaari pyrkii joka vuosi korostamaan luennoillaan uusia asioita ja tuomaan esille ajan trendejä. Tämän vuoden teemaksi valikoituivat itse tekeminen sekä modaus. Suosituimmiksi luennoiksi Ängeslevä katsoi ne, jotka sisälsivät eniten konkretiaa, sekä ne, joissa oli täysin uudenkarheaa sisältöä.

**Arja Martikainen** piti vaikuttavan luennon pelialan rekrytoinnista ja vastasi moniin aloittelijoita poltteleviin kysymyksiin.

”Vaikka Assemblyillä nykyään enimmäkseen pelataan, ihmisiä halutaan myös herätellä siihen, että pelialasta voi tehdä itselleen palkallisen ammatin. Jo ensimmäisestä osallistumiskerrasta voi syttyä kipinä. Jos kehittää omaa potentiaaliaan, voivat yrityksetkin kiinnostua ja yksi asia helposti johtaa toiseen.”

Assemblyillä tapahtuneiden kohtaamisten perusteella on tehty myös

suomalaista pelihistoriaa. Muun muassa menestysfirma Rovion perustajat **Kim Dikert, Jarno Väkeväinen** ja **Niklas Hed** löysivät toisensa 14 vuotta sitten kesä-Assemblyilta.

Martikainen toivoi Assemblyjen edelleen toimivan kohtauspaikkana erityisesti nuorille, joille tärkeintä on ensimmäinen merkki CV:ssä. Pelialan rekrytoinneissa korostuvat osaamisen lisäksi myös hakijan henkilökohtainen innostus ja harrastuneisuutta alaa kohtaan. Assemblyille osallistuminen on varteenotettava keino lihottaa omaa portfolioaan sekä osoittaa kiinnostustaan. Kokemuksen ja ammattilaisten vinkkien avulla saattaa kirvota idea myös oman firman perustamisesta.

### Toisiaan täydentäen

Pelin kaari ideasta valmiiksi tuotteeksi on pitkä prosessi. Harva hallitsee pelikehityksen prosessin kaikkia osia omin nokkineen, ja Assembly pyrkiikin tuomaan yhteen ihmisiä, joiden taidot täydentävät toisiaan. Peliala tarvitsee monipuolista osaamista aina taiteen tekemisestä piinkovaan projektihallintaan. Monille oma juttu löytyy helposti, mutta Assemblyt rohkaisevat etenkin vanhempia konkareita kokeilemaan pelikehityksen osa-alueita myös oman mukavuusalu-



Sonja Ängeslevä ja Arja Martikainen kehuvat Assemblyä innoittavaksi tapahtumaksi.

een ulkopuolelta.

”Erityisesti peliteknologian ensimmäisen aallon aikaan 90-luvulla oli yleistä hakeutua kansainvälisiin projekteihin suomalaisen pelialan kehittymättömyyden vuoksi. Tänä päivänä monet menestyneet suomalaiset pelifirmat lisäävät onnistuneesti alan kotimaisen osaamisen tunnettua”, Ängeslevä kertoo.

Myös Martikainen suitsutti Suomen kansainvälistä menestystä ja kertoo pelialan olevan yksi teknologian kärki-alueita, joilla alan innovaattoreita pyritään houkuttelemaan myös Suomeen.

”Jos kansainväliset pelikehittäjät pitävät Suomea potentiaalisena kohteena, myös heidän Suomi-kuvansa paranevat”, hän toteaa.

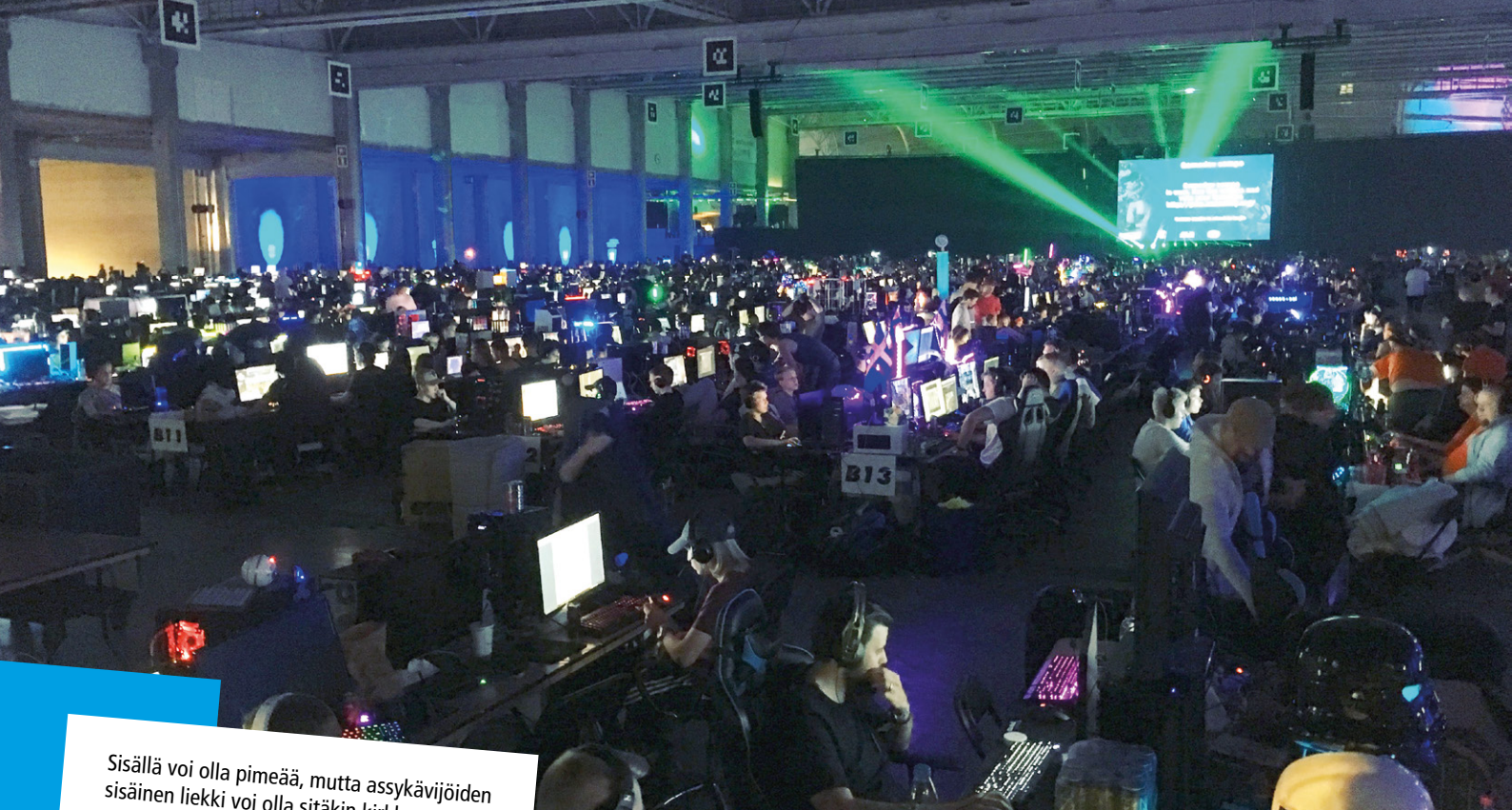
### Assembly yhdistää

Myös Small Giant Gamesin johtava pelisuunnittelija **Tim Lönnqvist** puhui seminaarissa. Hän kertoi pelikehityksen tuntuneen nuorena utopistiselta urapolulta ja päätyneensä alalle vahingossa. Harrastuspuoli ja kompoihiin osallistuminen jäi aikoihin väliin, koska IT-insinööriksi

Digikulttuurin kasvaessa entisestään suuri osa työmarkkinoista suuntautuu edelleen ulkomaille.







Sisällä voi olla pimeää, mutta assykvijöiden sisäinen liekki voi olla sitäkin kirkaampi.

opiskeltuaan hän pääsi graafiseksi suunnittelijaksi erilaisiin projekteihin. Portti pelimaailmaan avautui osaamisen kehittyessä. Ajalleen tyypillisen kaavan mukaan Lönnqvist aloitti uransa Englannista ja päätyi sen jälkeen takaisin kotimaahansa Suomeen.

Useissa pelifirmoissa työskennellyt Lönnqvist kertoi lähes kaikilla työtovereillaan ja useimmilla suomalaisilla ohjelmoijilla olevan Assembly-historiaa.

”Sillä on ollut merkitystä, että viimeiset 25 vuotta Suomessa on ollut paikka, jonne peliohjelmoijat voivat mennä riehumaan kerran vuodessa.”

Tapahtuma yhdistää myös laajalla spektrillä eri sukupolvia keskenään: ”Ensin sieltä on aloittanut itse omien työkavereiden kanssa ja myöhemmin sinne on vienyt omia lapsiaan.”

”Assembly on ollut jo oman sukupolveni teini-ikäisille tietokoneharrastajille suuri tapahtuma, jossa pääsee tapaamaan oman alan sankareita, joita ei välttämättä ole kaikissa vastaavissa tapahtumissa tai edes kaikissa maissa.”

Pelialalla ei yleensä tuijoteta liikaa papereita vaan haetaan yksilöitä, joilla on aitoa kiinnostusta alaa kohtaan sekä halua kehittyä. Lönnqvistin tapaan pelinteko on muuttunut monilla harrastuksesta ammatiksi. Aiemmin alan koulutus oli varsin kiven alla, mutta nykyään tarjonta on monipuolistunut merkittävästi ja eri osa-alueita

voi opiskella jopa yliopistotasolla. Tulvat työnantajat ovat kuitenkin eniten kiinnostuneita vankasta harrastuneisuudesta, innostuksesta alaa ja itseään kehittämistä kohtaan sekä mahdollisista meriteistä CV:ssä, joita pääsee keräämään muun muassa Assemblyltä.

”Suomessa pienet tiimit tekevät monimutkaisia pelejä. Joku voisi sanoa Suomessa olevan globaalisti poikkeuksellisen taitavia, monilahjakkaita pelintekijöitä, jotka suoriutuvat yksin laaja-alaisistakin töistä. Toisissa maissa, kuten vaikka Amerikassa ja Japanissa, yksilön ei välttämättä tarvitse hallita niin montaa osa-aluetta niin nuorena. Suomessa pelialan avoimet paikat saattavat olla välillä todella erikoisia ja spesifejä, ja etenkin pienissä firmoissa työntekijöiden moniulotteiset taidot osottautuvat käteviksi.”

Lönnqvist korosti erityisesti Assemblyiltäkin pelijamien merkitystä ja kertoi samanlaista menetelmää sovellettavan toisinaan myös yksityisissä pelifirmoissa luovuuden ja kokeellisuuden kehittämiseksi. Assembly-jameilla tehdyt pelit julkaistaan, mutta Lönnqvist kehuu vastaavasti tuloslähtöisyydettömyyden positiivista merkitystä firmojen sisäisen toiminnan kannalta.

”Jameissa pääsee kokeilemaan asioita, joita on miettinyt pitkään ja halunnut päästä kehittämään. Tuloslähtöisyys ei tarkoita, etteikö pelejä tulisi firmojenkin sisällä - päivittäin. Jammailu on tapa fokuroida työntekijöitä ja antaa erilaisia välineitä uusien menestysideoiden saavuttamiseksi.”

## Assemblylle toivotaan valoisaa tulevaisuutta

25 vuoden aikana niin peliteknologia kuin Assemblyt ovat nähneet paljon erilaisia vaiheita ja muuttuneet ajan mukana. Alusta asti mukana ollut demoscene on saanut rinnalleen peliskenen niin pelaamisen kuin pelintekonkin näkökulmasta, samaan aikaan kun pelaaminen on valloittanut valtaviiran. Teknologian riemuvoiton myötä pelaamisesta on tullut yleinen harrastus, joka kuuluu kaikille. Siinä missä nuorille tapahtumassa riittää aina uusia kiinnostuksen kohteita, syvä harrastuneisuus ja halu kilpailla kompoissa vetävät vanhojakin konkareita puoleensa.

Ängeslevä, Martikainen ja Lönnqvist kaikki toivoivat Assemblyjen jatkavan toimintaansa ja säilyvän kohtauspaikkana alan ihmisille uusia menestystarinoita varten.

”Toivottavasti Assembly jatkaa toimintaansa ja säilyy nuorten pelintekijöiden innoittajana. Pelit ovat tällä hetkellä joka paikassa ja monet menevät Assemblylle vain pelaamaan. Sieltä voi kuitenkin saada kosketuksen myös pelintekemisen, modaamisen, pelien muokkaamisen ja monen muun asian maailmaan. Kipinän voi saada yhdellä vilkaisulla. Tärkeintä on mennä rohkeasti kokeilemaan ja pitää kynns matalalla, koska vaikka se on iso pimeä sali, siellä ei ole mitään pelättävää.” 🎮



ENNAKKOHINTA – VIELÄ EHTII

# TILAA SKROLLIN VUOSIKERTA 2018 PAPERILLA TAI DIGINÄ

# 35 €

**SKROLLI**

[tilaaskrolli.fi](http://tilaaskrolli.fi)

# QUIT HORSING AROUND



## Join Reaktor

Forget unicorns – we'd much rather work with real human beings.  
Don't be ready. Be curious to learn. Apply today.

[reaktor.com/careers](https://reaktor.com/careers)

